

Higher order Hamiltonian Monte Carlo sampling for cosmological large-scale structure analysis

Mónica Hernández-Sánchez,^{1,2★} Francisco-Shu Kitaura^{1,2★}, Metin Ata³ and Claudio Dalla Vecchia^{1,2}

¹*Instituto de Astrofísica de Canarias (IAC), Calle Vía Láctea s/n, E-38200 La Laguna, Tenerife, Spain*

²*Departamento de Astrofísica, Universidad de La Laguna (ULL), E-38206 La Laguna, Tenerife, Spain*

³*Kavli Institute for the Physics and Mathematics of the Universe (Kavli IPMU), WPI, UTIAS, The University of Tokyo, Kashiwa, Chiba 277-8568, Japan*

Accepted 2021 January 9. Received 2021 January 9; in original form 2020 December 15

ABSTRACT

We investigate higher order symplectic integration strategies within Bayesian cosmic density field reconstruction methods. In particular, we study the fourth-order discretization of Hamiltonian equations of motion (EoM). This is achieved by recursively applying the basic second-order leap-frog scheme (considering the single evaluation of the EoM) in a combination of even numbers of forward time integration steps with a single intermediate backward step. This largely reduces the number of evaluations and random gradient computations, as required in the usual second-order case for high-dimensional cases. We restrict this study to the lognormal-Poisson model, applied to a full volume halo catalogue in real space on a cubical mesh of $1250 h^{-1}$ Mpc side and 256^3 cells. Hence, we neglect selection effects, redshift space distortions, and displacements. We note that those observational and cosmic evolution effects can be accounted for in subsequent Gibbs-sampling steps within the COSMIC BIRTH algorithm. We find that going from the usual second to fourth order in the leap-frog scheme shortens the burn-in phase by a factor of at least ~ 30 . This implies that 75–90 independent samples are obtained while the fastest second-order method converges. After convergence, the correlation lengths indicate an improvement factor of about 3.0 fewer gradient computations for meshes of 256^3 cells. In the considered cosmological scenario, the traditional leap-frog scheme turns out to outperform higher order integration schemes only when considering lower dimensional problems, e.g. meshes with 64^3 cells. This gain in computational efficiency can help to go towards a full Bayesian analysis of the cosmological large-scale structure for upcoming galaxy surveys.

Key words: methods: data analysis – methods: numerical – methods: statistical – large-scale structure of Universe – cosmology: observations.

1 INTRODUCTION

In the current cosmological picture, the nonlinear structures we observe today have risen from some closely Gaussian primordial fluctuations (see e.g. Mo, van den Bosch & White 2010, and references therein). Gaussian fields have the convenient property of being fully characterized by the variance, i.e. the two-point statistics, which is given by the correlation function in configuration space, or the power spectrum in Fourier space. It is thus common to extract cosmological information from the two-point statistics (see e.g. Beutler et al. 2017; Chuang et al. 2017; Ross et al. 2017). However, as gravity couples different scales, the cosmic density field is far from being Gaussian anymore, and the linear predictions of the two-point statistics do not match the observations (see e.g. Libeskind et al. 2018). Therefore, nonlinear models have been developed to be able to compare the theoretical predictions to the observations and constrain cosmological parameters (see e.g. Angulo et al. 2008; Nishimichi et al. 2009; Reid & White 2011; Okumura et al. 2015; Uhlemann & Kopp 2015; White 2015; Bose & Koyama 2017; Hashimoto,

Rasera & Taruya 2017). But even if one succeeds in doing so, not all the cosmological information is encoded in the two-point statistics in low redshift data, as opposed to the cosmic microwave background (see e.g. Schmittfull et al. 2015). This is why linearization methods have been suggested in the literature (Neyrinck, Szapudi & Szalay 2009; Kitaura & Angulo 2012; Schuhmann, Joachimi & Peiris 2016). In particular, reconstruction takes the galaxies back in time, putting back information from the higher order into the two-point statistics, thus increasing the precision of baryon acoustic oscillation (BAO) signature measurement (Eisenstein et al. 2007; Padmanabhan et al. 2012). Other ways of gaining nonlinear information from the galaxy distribution have been suggested based on the three-point statistics (see e.g. Saito et al. 2014; Gil-Marín et al. 2017), or on reconstructions of cosmic voids (Kitaura et al. 2016; Zhao et al. 2020). From a Bayesian perspective, one can write the posterior distribution function relating the primordial density field to the galaxy distribution through a Gaussian prior and some likelihood including nonlinear dynamics and some bias description (Kitaura & Enßlin 2008). The resulting global posterior probability distribution function (PDF) is clearly non-Gaussian. One of the simplest models we can consider is the lognormal-Poisson, accounting for the non-Gaussian matter distribution and the discreteness of the galaxy distribution (Kitaura, Jasche & Metcalf 2010). More complex variations on

* E-mail: mhs@iac.es (MH-S); fkitaura@iac.es (F-SK)

this can be suggested, including deviations from Poissonity in the likelihood, or nonlinear dynamics in the connection between the initial and final cosmic density field. As a matter of fact, the lognormal-Poisson model can be an accurate model for Lagrangian tracers, which are connected within a Gibbs-sampling scheme to the observed galaxy field distribution sampling the displacements in a separated step (Kitaura et al. 2020). Hamiltonian Monte Carlo techniques permit us to sample from non-Gaussian PDFs (Duane et al. 1987; Neal 1993; Jasche & Kitaura 2010; Neal 2012). Ever since the first application to observational data from galaxy surveys without (Jasche et al. 2010) and with cosmic evolution modelling (Kitaura et al. 2012b), a number of Bayesian inference methods have been developed to solve the problem of sampling linear density fields from a galaxy distribution (Jasche & Wandelt 2013; Kitaura 2013; Wang et al. 2013, 2014; Bos, Kitaura & van de Weygaert 2019; Jasche & Lavaux 2019). However, one of the drawbacks of these methods is that they require thousands of accepted iterations until convergence, and have very long correlation lengths of several hundred to one thousand iterations. As galaxy surveys increase in volume, accurate reconstructions demand meshes with between 100 million to one billion cells. Given the high dimensionality of the problem, Bayesian methods cannot be considered yet to be practical to sample full posterior distributions and constrain cosmological parameters.

This calls for efforts in increasing the efficiency of the Hamiltonian Monte Carlo Sampling. A number of works have investigated higher order discretizations of the Hamiltonian equations of motions (for a comprehensive summary see Hairer, Lubich & Wanner 2010). Yoshida (1990) proposed a higher order symplectic integration parametrizing the integration steps and calculating the exact coefficients. Also, efforts have been done in the field of quantum-chromodynamics and lattice computations, successively applying second-order leap-frog integrations (Creutz 1988; Creutz & Gocksch 1989; Camprostrini & Rossi 1990; Kennedy 2006; Luscher 2010). In the field of applied mathematics, Blanes, Casas & Sanz-Serna (2014) suggested a higher order integrator by sampling from Gaussian distributions and splitting the integration scheme, evaluating the force term several times per integration step. This has been incorporated into a general N -body integration framework in Rein & Tamayo (2018). Multisymplectic integrators (e.g. Islas & Schober 2004) are extensively used describing the evolution of the Schrödinger equation in quantum field theory. For other advances in higher order symplectic integration methods see Omelyan, Mryglod & Folk (2002).

Also other works have investigated higher order discretization schemes (Chao et al. 2015; Mannseth, Kleppe & Skaug 2016; Barp et al. 2018), in particular in the field of the cosmic microwave background (Taylor et al. 2008; Souradeep, Das & Wandelt 2016), however, without aiming at accelerating the Hamiltonian sampling method, or with very little success in this aspect.

In this work, we investigate the computational efficiency of the fourth-order leap-frog scheme by recursively applying the basic second-order leap-frog scheme in a combination of even numbers of forward time integration steps with a single intermediate backward step, following the works of (Creutz 1988; Creutz & Gocksch 1989; Camprostrini & Rossi 1990). It is important to stress, that, instead of applying the second-order leap-frog scheme as it is usually done within the recursive formula, we consider only the single evaluation of the Hamiltonian equations of motion. This means that we do not include the randomization of the number of steps for fourth-order scheme. In this way, we can squeeze the potential of the larger integration steps allowed by the higher order scheme, as we show

in this paper. Recent studies indicate that clever applications of recursive forward and backward second-order leap-frog computations can yield significant efficiency improvements in high dimensional spaces (see the NUTS scheme Hoffman & Gelman 2014, showing improvements of up to factors of 3). Generally, higher accurate symplectic integrators are computationally more expensive, thus it is important to find a beneficial trade-off for computational costs and gains in phase-space movement. We show in this work that higher order integrators can become significantly more efficient in high statistical dimensions, as expected from mathematical considerations (see Beskos et al. 2013; Bou-Rabee & Sanz-Serna 2018). For higher order methods solving classical Hamiltonian systems see McLachlan (2002). This field of research indicates that more sophisticated higher order schemes than the one studied in this paper are very promising for Bayesian studies (see McLachlan & Quispel 2002).

The work presented here potentially represents a major step forward in Bayesian inference studies within cosmological large-scale structure analysis. This paper is a companion paper of Kitaura et al. (2020).

The remainder of this manuscript is structured as follows, first we revise the theory of Hamiltonian Monte Carlo sampling and present the higher order formalism. Then we describe the data used in this work and the numerical tests performed on them. Finally we present our summary and conclusions.

2 METHOD

For the sake of completeness we will recap the lognormal-Poisson posterior model within a Bayesian framework, first presented in Kitaura et al. (2010). In particular, we will include a power-law bias description in the equations as introduced in Ata, Kitaura & Müller (2015).

2.1 Bayesian framework

In a Bayesian inference framework we need to first define the prior of the sought signal s : $\pi(s)$, and then the likelihood of the data given the signal $\mathcal{L}(d|s)$. These ingredients permit us to define the posterior distribution function, i.e. the PDF of a signal given the data:

$$\mathcal{P}(s|d) \propto \pi(s) \times \mathcal{L}(d|s) \quad (1)$$

2.1.1 The prior

The signal we want to reconstruct within a Bayesian inference framework is the linear over-density field, thus, $s \equiv \delta_L$. From now on, we will consider a regular grid with a cubical volume V of side L subdivided into N_c cells. We assume as a prior that δ_L is Gaussian distributed with zero mean

$$\pi(\delta_L | \mathbf{C}_L) = \frac{1}{\sqrt{(2\pi)^{N_c} \det(\mathbf{C}_L)}} \exp\left(-\frac{1}{2} \delta_L^T \mathbf{C}_L^{-1} \delta_L\right), \quad (2)$$

where $\mathbf{C}_L = \langle \delta_L^T \delta_L \rangle$ is the co-variance matrix.

2.1.2 The likelihood

The likelihood defines the model of the data. One has to include here the connection between the signal s and the data d . This is achieved with a structure formation model for the dark matter field, relating the primordial linear over-density field δ_L to the cosmic evolved one δ , and a biasing prescription relating δ to the galaxy population.

The dark matter density field

In this work, we relate the nonlinear over-density field $\delta = \rho/\bar{\rho} - 1$ (with ρ being the density) through a logarithmic transformation to linear

$$\delta_L = \log(1 + \delta) - \mu, \quad (3)$$

where

$$\mu = \langle \log(1 + \delta) \rangle. \quad (4)$$

This yields the lognormal model for the density field. It is particularly interesting due to its rich cosmological information content (Carron & Szapudi 2014). Looking carefully at its derivation from the continuity equation applied to a cosmic fluid, one finds that it is valid for a Lagrangian co-moving framework before shell crossing (Coles & Jones 1991; Kitaura & Angulo 2012). This implies that the lognormal assumption, applied to cosmic evolved density fields observed in Eulerian coordinates, is not accurate, especially in the three-point statistics (White, Tinker & McBride 2014; Chuang et al. 2015), although it gives a fair description of the two-point statistics (Neyrinck et al. 2009). Nevertheless, we will use this prior in this work as a reference to study the efficiency of the sampler, neglecting displacements connecting Lagrangian to Eulerian space, as it would be required for an accurate structure formation description. We note, however, that more complex structure formation models can be implemented. Sampling the Lagrangian tracers with the displacement field given an arbitrary structure formation model within a Gibbs-sampling framework, the lognormal assumption turns out to become reasonable, and for $|\delta| \ll 1$ it ultimately tends towards the Gaussian PDF (Kitaura et al. 2020). The lognormal assumption, however, ensures positive definite densities, i.e. $\rho \geq 0$. This is very important, since otherwise one has to cut-off within a general nonlinear bias description cells with $\rho < 0$ resulting in an artificial lack of power of the density field.

Link between the dark matter field and the galaxy distribution

It is natural to define the data as the number counts of galaxies $\mathbf{d} \equiv \mathbf{N}^g$ in the above defined regular grid, as this allows for a clear statistical description, and for efficient operations relying on fast Fourier transforms.

To capture the discrete nature of the data we can assume a Poisson likelihood, which was introduced to Bayesian reconstruction in cosmology in Kitaura & Enßlin (2008), Kitaura et al. (2010)

$$\mathcal{L}(\mathbf{N}_k^g | \lambda_k) = \prod_k \frac{(\lambda_k)^{N_k^g} e^{-\lambda_k}}{N_k^g!}, \quad (5)$$

where the expected number counts per cell k is given by

$$\lambda_k = f_N w_k (1 + \delta_k)^b, \quad (6)$$

f_N is the normalization of the ensuring a given number density \bar{N} , w_k is the three-dimensional completeness at cell k , and b is the power-law bias parameter (see Kitaura, Yepes & Prada 2014; Ata et al. 2015).

Prior reconstructions considered only the variance of the Poisson distribution within a Gaussian likelihood, which does not ensure positive definite density fields in the reconstruction (Zaroubi et al. 1995). We note that this is the simplest discrete PDF we can consider without requiring any additional parameter. In general, the distribution of galaxies is not Poisson distributed (see Peebles 1980). There are some PDFs which can capture the deviation from

Poissonity (see e.g. Saslaw 1989; Sheth 1998; Kitaura et al. 2014; Neyrinck et al. 2014; Ahn et al. 2015), that can be implemented in a Bayesian framework (Ata et al. 2015).

2.1.3 The posterior

Based on the prior and likelihood defined in the previous sections we can now define the posterior PDF (see equation 1). For convenience, let us write the negative logarithm of the posterior as

$$-\ln \mathcal{P} = -\ln \pi - \ln \mathcal{L}. \quad (7)$$

This permits us to write the prior term 2 as

$$-\ln \pi(\delta_L | \mathbf{C}_L) = \frac{1}{2} \delta_L^T \mathbf{C}_L^{-1} \delta_L + c, \quad (8)$$

where we have included terms that do not depend on the signal in the term c . The negative logarithm of the likelihood, taking equation (5), is simply

$$-\ln \mathcal{L}(\mathbf{N}_k^g | \lambda_k) = \sum_k \lambda_k - N_k \ln \lambda_k + c', \quad (9)$$

with $c' \neq c'(\delta_L)$. This permits us to compute the gradients with respect to the signal of the prior and the likelihood in a straightforward way as introduced in Kitaura et al. (2010). For the prior we obtain

$$-\frac{\partial \ln \pi}{\partial \delta_L} = \mathbf{C}_L^{-1} \delta_L. \quad (10)$$

And for the likelihood we use the chain rule to get

$$-\frac{\partial \ln \mathcal{L}}{\partial \delta_{L,i}} = b(\lambda_i - N_i). \quad (11)$$

These gradients permit us to compute either the maximum a posteriori, when solving the corresponding equation set to zero, or to sample from the posterior PDF using the Hamiltonian Monte Carlo sampling, as we will show in the next section.

2.2 Hamiltonian Monte Carlo sampling

To sample the posterior we rely on the Hamiltonian Monte Carlo sampling technique (HMC; Duane et al. 1987; Neal 1993; Jasche & Kitaura 2010). Let us recap the method in this section and extend it to higher orders. The Hamiltonian is defined as a function of the generalized phase space coordinates of positions \mathbf{q} and momenta \mathbf{p} , through the potential energy $\mathcal{U}(\mathbf{q})$ and the kinetic energy $\mathcal{K}(\mathbf{p})$

$$\mathcal{H}(\mathbf{q}, \mathbf{p}) = \mathcal{U}(\mathbf{q}) + \mathcal{K}(\mathbf{p}). \quad (12)$$

The kinetic energy is expressed as

$$\mathcal{K}(\mathbf{p}) = \frac{1}{2} \mathbf{p}^T \mathbf{M}^{-1} \mathbf{p}, \quad (13)$$

where \mathbf{M} is the mass matrix, describing the co-variance of the momenta. It represents the degree of freedom in the Hamiltonian sampler, and its structure can be crucial for the efficiency (Neal 2012). One chooses an adequate mass matrix, encoding both the prior and the likelihood information. In general, such a mass matrix will be non-diagonal, and there are ways of implementing them in an efficient way (Kitaura et al. 2020). In this work we will restrict our studies to a full volume, for which a mass matrix given by the inverse matter co-variance matrix is nearly optimal, $\mathbf{M} = \mathbf{C}_L^{-1}$ (see Taylor, Ashdown & Hobson 2008).

To relate the Hamiltonian dynamics to a probabilistic measure, we resort to the canonical distribution definition

$$\mathcal{P}(\mathbf{q}, \mathbf{p}) = \frac{1}{Z} e^{-\mathcal{H}(\mathbf{q}, \mathbf{p})}, \quad (14)$$

where Z is the normalization of the distribution function. The latter equation can also be expressed as

$$\mathcal{P}(\mathbf{q}, \mathbf{p}) = \mathcal{P}(\mathbf{q})\mathcal{P}(\mathbf{p}) = \frac{1}{Z} e^{-\mathcal{U}(\mathbf{q})} e^{-\mathcal{K}(\mathbf{p})}, \quad (15)$$

according to our previous definitions, factorized into two separated probabilities corresponding to the potential energy (and the positions): $\mathcal{P}(\mathbf{q})$, and to the kinetic energy (and the momenta): $\mathcal{P}(\mathbf{p})$. It is interesting now to identify the potential energy $\mathcal{U}(\mathbf{q})$ with the negative logarithm of the posterior distribution function (equation 7)

$$\mathcal{U}(\mathbf{q}) = -\ln \mathcal{P}(\mathbf{q}), \quad (16)$$

and realize that the kinetic term $\mathcal{K}(\mathbf{p})$ defines a multivariate Gaussian distribution function

$$\mathcal{P}(\mathbf{p}) \propto e^{-\mathcal{K}} = e^{-\frac{1}{2} \mathbf{p}^T \mathbf{M}^{-1} \mathbf{p}}. \quad (17)$$

This implies that the Hamiltonian Monte Carlo sampling only requires a Gaussian field with a free Hamiltonian mass to sample arbitrary non-Gaussian PDFs. We can now further identify the positions, \mathbf{q} , as the variable to sample, i.e. the sought signal, in our case, the primordial fluctuations δ_L . The momenta, \mathbf{p} , are artificially introduced in the kinetic term just to allow us to explore the phase-space, therefore, to evolve the system and get \mathbf{q} . The marginalization is done to avoid the dependence on the momenta when obtaining the posterior. This is achieved by randomly drawing new momenta in each iteration, disregarding the ones of the previous step.

The partial derivatives of the Hamiltonian determine how \mathbf{q} and \mathbf{p} change with time, t , according to Hamilton's equations of motion

$$\frac{d\mathbf{q}}{dt} = \frac{\partial \mathcal{H}(\mathbf{q}, \mathbf{p})}{\partial \mathbf{p}} = \{\mathcal{H}, \mathbf{q}\}, \quad (18)$$

$$\frac{d\mathbf{p}}{dt} = -\frac{\partial \mathcal{H}(\mathbf{q}, \mathbf{p})}{\partial \mathbf{q}} = -\{\mathcal{H}, \mathbf{p}\}, \quad (19)$$

where we have introduced the Poisson bracket definition: $\{f, g\} = \left(\frac{\partial f}{\partial \mathbf{q}}\right)^T \frac{\partial g}{\partial \mathbf{p}} - \left(\frac{\partial f}{\partial \mathbf{p}}\right)^T \frac{\partial g}{\partial \mathbf{q}}$, for reasons which will be clear below. Taking into account equations (12) and (13), Hamilton's equations can then be written as

$$\frac{d\mathbf{q}}{dt} = \mathbf{M}^{-1} \mathbf{p}, \quad (20)$$

$$\frac{d\mathbf{p}}{dt} = -\frac{\partial \mathcal{U}(\mathbf{q})}{\partial \mathbf{q}}. \quad (21)$$

The particular expression to the latter equation, in our case study, is given by the sum of equations (10) and (11).

Moreover, the Hamiltonian dynamics has to fulfil a series of properties:

(i) The Hamiltonian \mathcal{H} is conserved as \mathbf{q} and \mathbf{p} evolve through time: $\frac{\partial \mathcal{H}}{\partial t} = 0$.

(ii) The dynamics also preserves the phase space volume according to Liouville's theorem.

(iii) Hamiltonian dynamics is reversible, i.e. mapping from a state to the next state is bijective one-to-one, and therefore, the inverse mapping is obtained by changing the sign in the time derivatives in equations (18) and (19).

These properties together imply that the canonical distribution is invariant with respect to any transformation.

2.2.1 Discretization and efficiency

To evolve the Hamiltonian system numerically, we must discretize the Hamilton's equations of motion using a finite time step, and

thus introducing an inevitable error. An explicitly time-invariant Hamilton function is energy conserving. Thus, the Hamiltonian difference $\Delta \mathcal{H}(\mathbf{q}, \mathbf{p}) = \mathcal{H}(\mathbf{q}', \mathbf{p}') - \mathcal{H}(\mathbf{q}, \mathbf{p})$ between the old (\mathbf{q}, \mathbf{p}) and new $(\mathbf{q}', \mathbf{p}')$ proposed state of the system should vanish. Nonetheless, numerical errors in the discretization scheme violate this conservation.

Due to this numerical error, one has to introduce a Metropolis–Hastings rejection step. The proposed new state obtained by

$$\mathcal{P}_{\text{acceptance}} = \min [1, e^{-\Delta \mathcal{H}(\mathbf{q}, \mathbf{p})}] , \quad (22)$$

where $\Delta \mathcal{H}(\mathbf{q}, \mathbf{p}) = \mathcal{H}(\mathbf{q}', \mathbf{p}') - \mathcal{H}(\mathbf{q}, \mathbf{p})$ stands for the difference in the Hamiltonian between the old (\mathbf{q}, \mathbf{p}) and new $(\mathbf{q}', \mathbf{p}')$ proposed state of the system.

The chosen time-step and discretization scheme will have a great impact on the acceptance rate and the computational efficiency.

Since the energy is an extensive quantity, the total error will grow with an increasing number of dimensions (see Beskos et al. 2013; Bou-Rabee & Sanz-Serna 2018). For this reason, for a given error per dimension, there is a number of dimensions from which on higher order schemes become more efficient than the standard second-order discretization scheme allowing for larger step sizes with higher acceptance rates. We will therefore investigate our physical problem at different resolutions in Section 3. Let us revise these schemes first from a theoretical perspective.

2.2.2 Second-order discretization

Let us follow the formalism of Creutz (1988), Creutz & Gocksch (1989), Camprostrini & Rossi (1990). We start with the basic leap-frog algorithm. For a Hamiltonian of the generalized coordinates \mathbf{q}, \mathbf{p} : $\mathcal{H}(\mathbf{q}, \mathbf{p})$, we can define the translation operator $\mathcal{T}(\epsilon)$, evolving the system along a time step of size ϵ . Due to the property listed in the previous subsection on reversibility of the Hamiltonian, we demand

$$\mathcal{T}^{-1}(\epsilon) = \mathcal{T}(-\epsilon). \quad (23)$$

We can now split the time translation into separated parts acting on \mathbf{p} and \mathbf{q} individually

$$\mathcal{T}_q(\epsilon) : (\mathbf{q}, \mathbf{p}) \rightarrow (\mathbf{q}', \mathbf{p}), \quad (24)$$

$$\mathcal{T}_p(\epsilon) : (\mathbf{q}, \mathbf{p}) \rightarrow (\mathbf{q}, \mathbf{p}'), \quad (25)$$

where the new states \mathbf{q}' and \mathbf{p}' are given according to the equations of motion (20) and (21) with

$$\mathbf{q}' = \mathbf{q} + \epsilon \mathbf{M}^{-1} \mathbf{p}, \quad (26)$$

$$\mathbf{p}' = \mathbf{p} - \epsilon \frac{\partial \mathcal{U}}{\partial \mathbf{q}}. \quad (27)$$

Following equations (18) and (19) the evolution of system from an old state (\mathbf{q}, \mathbf{p}) to a new one $(\mathbf{q}', \mathbf{p}')$ is obtained through the action of the Hamilton operator on (\mathbf{q}, \mathbf{p}) . A naive translation of step size $\Delta \tau = \epsilon$, such as $\mathcal{T}(\epsilon) = \mathcal{T}_p(\epsilon)\mathcal{T}_q(\epsilon)$, will violate time reversibility since

$$(\mathcal{T}_p(\epsilon)\mathcal{T}_q(\epsilon))^{-1} = \mathcal{T}_q(\epsilon)^{-1}\mathcal{T}_p(\epsilon)^{-1} \neq \mathcal{T}_p(\epsilon)^{-1}\mathcal{T}_q(\epsilon)^{-1}. \quad (28)$$

An obvious choice of a time translation operator to preserve reversibility can be constructed by symmetrizing the operator

$$\mathcal{T}_2(\epsilon) = \mathcal{T}_p(\epsilon/2)\mathcal{T}_q(\epsilon)\mathcal{T}_p(\epsilon/2), \quad (29)$$

which is the commonly used leap-frog discretization scheme. It preserves phase space volume, and is also time reversible. We favour this formulation with respect to $\mathcal{T}_q(\epsilon/2)\mathcal{T}_p(\epsilon)\mathcal{T}_q(\epsilon/2)$ (interchanging the role of p with q). It permits us to save some matrix inversions when computing $\mathbf{M}^{-1}\mathbf{p}$.¹ This becomes clear further below (whether, despite of these extra computations, this formulation is more efficient needs to be investigated in future work). A single iteration calculates approximations to the position and momenta at time $t + \epsilon$ from these quantities at t as it follows:

$$\mathbf{p}\left(t + \frac{\epsilon}{2}\right) = \mathbf{p}(t) - \frac{\epsilon}{2} \frac{\partial \mathcal{U}}{\partial \mathbf{q}}(\mathbf{q}(t)) \quad (30)$$

$$\mathbf{q}(t + \epsilon) = \mathbf{q}(t) + \epsilon \mathbf{M}^{-1} \mathbf{p}\left(t + \frac{\epsilon}{2}\right) \quad (31)$$

$$\mathbf{p}(t + \epsilon) = \mathbf{p}\left(t + \frac{\epsilon}{2}\right) - \frac{\epsilon}{2} \frac{\partial \mathcal{U}}{\partial \mathbf{q}}(\mathbf{q}(t + \epsilon)), \quad (32)$$

where the last gradient computation (equation 32) serves as the first one (equation 30) for the next iteration. Note that the momenta are replaced after each completed Hamiltonian MC iteration, and therefore the last matrix inversion applied to the momenta cannot always be used for the next iteration in a scheme of the form $\mathcal{T}_q(\epsilon/2)\mathcal{T}_p(\epsilon)\mathcal{T}_q(\epsilon/2)$. Equations (30) to (32) correspond to a second-order discretization of the equations of motion, ($\mathcal{O}(\epsilon^2)$), as we will discuss in the next section. Equations (30)–(32). In practice, this scheme is applied

$$N_{\text{eval}}^{\text{2nd}} = 1 + \text{floor}(u_N \times N_{\text{eval}}) \quad (33)$$

times with a random time step $\epsilon^{\text{eff}} = u_\epsilon \times \epsilon$, where u_N and u_ϵ are random numbers, which help the HMC sampler to explore the parameter space avoiding resonant trajectories (Neal 1993). The corresponding second-order leap-frog scheme is thus given by multiple evaluations of the equations of motion

$$N_{\text{eval}}^{\text{2nd}} \times (\mathcal{T}_2(\epsilon^{\text{eff}}) = \mathcal{T}_p(\epsilon^{\text{eff}}/2)\mathcal{T}_q(\epsilon^{\text{eff}})\mathcal{T}_p(\epsilon^{\text{eff}}/2)) \quad (34)$$

Hence, the global step size is given by

$$\Delta \tau_{\text{2nd}} = N_{\text{eval}}^{\text{2nd}} \times \epsilon^{\text{eff}} = (1 + \text{floor}(u_N \times N_{\text{eval}})) \times u_\epsilon \times \epsilon. \quad (35)$$

2.2.3 Higher order discretization

Let us now revise Hamilton mechanics to find a generalization of the leap-frog integration beyond second-order. For any conserved function, which depends on the phase-space variables at time t , $f(t, \mathbf{q}, \mathbf{p})$, the time derivative vanishes

$$\frac{df}{dt} = 0, \quad (36)$$

and hence

$$\frac{\partial f}{\partial t} + \frac{\partial f}{\partial \mathbf{q}} \frac{d\mathbf{q}}{dt} + \frac{\partial f}{\partial \mathbf{p}} \frac{d\mathbf{p}}{dt} = \frac{\partial f}{\partial t} - \{\mathcal{H}, f\} = 0, \quad (37)$$

where we have used equations (18) and (19), and the Poisson bracket definition. Thus, we can identify the partial time derivative to the Hamilton operator

$$\frac{\partial}{\partial t} f = \{\mathcal{H}, \cdot\} f. \quad (38)$$

From this we can write a time evolution of f from time t to $t + \epsilon$ by following transformation

$$f(t + \epsilon, \mathbf{q}, \mathbf{p}) = e^{\mathcal{H}\epsilon} f(t, \mathbf{q}, \mathbf{p}), \quad (39)$$

¹Within the COSMIC BIRTH code this operation involves more operations (including convolutions) as explained in the appendix in Kitauro et al. (2020).

which is the classical equivalent to the time evolution solution for the Schrödinger equation in quantum mechanics. Expanding $f(t, \mathbf{q}, \mathbf{p})$ as a function of the time evolution in a Taylor series, we can write

$$\begin{aligned} f(t + \epsilon, \mathbf{q}, \mathbf{p}) &= f + \frac{\partial}{\partial t} f \epsilon + \frac{1}{2} \frac{\partial^2}{\partial t^2} f \epsilon^2 + \frac{1}{6} \frac{\partial^3}{\partial t^3} f \epsilon^3 + \dots, \\ &= f + \{\mathcal{H}, \cdot\} f \epsilon + \frac{1}{2} \{\mathcal{H}, \{\mathcal{H}, \cdot\}\} f \epsilon^2 + \\ &\quad \frac{1}{6} \{\mathcal{H}, \{\mathcal{H}, \{\mathcal{H}, \cdot\}\}\} f \epsilon^3 + \dots, \end{aligned} \quad (40)$$

where we have used equation (38). We can relate equation (39) to the series expansion given by equation (40) and write

$$e^{\mathcal{H}\epsilon} = \mathcal{T}_n(\epsilon) - \Delta_{n+1}\epsilon^{n+1} + \mathcal{O}(\epsilon^{n+2}), \quad (41)$$

where the errors of the time evolution operator \mathcal{T}_n , with respect to the analytic solution, $-\Delta_{n+1}\epsilon^{n+1}$, are of order $n + 1$. The operator \mathcal{T}_n is made out of a number of concatenated Hamiltonian operators depending on the order n . This suggests the idea to construct higher order Hamiltonian schemes based on recursive applications of lower order ones. A naive fourth-order scheme could then be constructed as the application of two successive second-order ones. Let us define with that spirit our tentative $(n + 2)$ -order operator as

$$\mathcal{T}_{n+2}(2i\epsilon) = \mathcal{T}_n^i(\epsilon)\mathcal{T}_n^i(\epsilon), \quad (42)$$

which based on equation (41) can be expanded to

$$\begin{aligned} \mathcal{T}_{n+2}(2i\epsilon) &= e^{\mathcal{H}2i\epsilon} + \Delta_{n+1}2i\epsilon^{n+1} + \\ &\quad \Delta_{n+2}\epsilon^{n+2} + \mathcal{O}(\epsilon^{n+3}), \end{aligned} \quad (43)$$

where i is the number of times the operator \mathcal{T}_n is successively applied, and Δ_m stands for the error factors at different order $m = 1, 2, \dots$. We have kept only the first term of the Taylor expansion of $e^{\mathcal{H}i\epsilon}$, whenever it appeared multiplying error terms to correctly keep track of the orders. This implies that the naive successive concatenation of second-order leap-frog operations does not yield a fourth-order accurate scheme. The problem here is the presence of error terms of order below $\mathcal{O}(\epsilon^{n+3})$. Let us focus first on the $(n + 1)$ -order term $\Delta_{n+1}2i\epsilon^{n+1}$. The solution proposed by Creutz & Gocksch (1989) and Campostrini & Rossi (1990) consists of introducing a backward step to exactly cancel out the $(n + 1)$ -order error term, which necessarily needs to have a step size of

$$-s\epsilon = -(2i)^{1/(n+1)}\epsilon. \quad (44)$$

From equation (41) we can verify that an opposite error term $-\Delta_{n+1}2i\epsilon^{n+1}$ is obtained. To see how the $(n + 2)$ -order term vanishes we need to construct a time-reversible operator (see equation 23), for which $\mathcal{T}(-\epsilon)\mathcal{T}(\epsilon) = 1$ holds within the order of the scheme. Inserting the expansions from equation (41) with the ansatz of equation (42) we get

$$\begin{aligned} \mathcal{T}_{n+2}(-\epsilon)\mathcal{T}_{n+2}(\epsilon) &= 1 + \\ &\quad \Delta_{n+1}(\epsilon^{n+1} - \epsilon^{n+1}) + \Delta_{n+2}(\epsilon^{n+2} - \epsilon^{n+2}) + \mathcal{O}(\epsilon^{n+3}). \end{aligned} \quad (45)$$

Focusing now on the $(n + 2)$ -order term we find that it can vanish for odd numbers of n , since then $\epsilon^{n+2} - \epsilon^{n+2}$ cancels out, however, with the term $\Delta_{n+2}\epsilon^{n+2}$ not having to be zero. Only for even numbers of n , and in order to accomplish the reversibility condition, we can state that Δ_{n+2} vanishes, and hence also the $(n + 2)$ -order error term. For this reason, Creutz & Gocksch (1989) suggested the following recursive scheme for even numbers of n

$$\mathcal{T}_{n+2}((2i - s)\epsilon) = \mathcal{T}_n^i(\epsilon)\mathcal{T}_n(-s\epsilon)\mathcal{T}_n^i(\epsilon), \quad (46)$$

Table 1. Deimos/Diva characteristics.

Use	Host	CPU	Freq	Cores	RAM	Disc
Login & GPU node	Deimos	2x Intel Xenon E5-2630 v4	2.20 GHz	20	1 TB	11 TB
Computing node	Diva	12x Intel Xenon E5-2630 v4	2.10 GHz	192	4.5 TB	40 TB

where reversibility and phase-space volume conservation are accomplished. Hence, iterating this scheme recursively produces a discretization of Hamilton's equations of motion to any desired even order. To make a fourth-order scheme one needs $2i + 1$ times computations of the basic second-order leap-frog scheme. The corresponding global step size is given by

$$\Delta\tau_{4th} = (2i - s(n = 2))\epsilon^{\text{eff}} = (2i - (2i)^{1/(3)})\epsilon^{\text{eff}}, \quad (47)$$

with a random time step $\epsilon^{\text{eff}} = u_\epsilon \times \epsilon$, as in the second-order case. It is important to stress that, applying the second-order leap-frog as usual, making large numbers of evaluations ($N_{\text{eval}}^{\text{eff}}$) of Hamiltonian equations of motion within the fourth-order method (equation 46), will result in an inefficient scheme. Instead, we explore in this work the application of this recursive formula calling the second-order discretization only $2i + 1$ times, and we explore the optimal i number. The aim of this is to explore the possibility of making larger time steps at a comparable computational cost, and hence gain efficiency sampling from the posterior PDF.

3 NUMERICAL VALIDATION

In this section we show our parameter study, analysing the optimal setting for the fourth-order leap-frog algorithm, as compared to the second-order discretization scheme. We start exploring the parameter space on a lower resolution, and then we focus on a number of constrained configurations on a set of higher resolution runs. Based on this we will make a robust assessment of the convergence of the chains and the corresponding correlation lengths.

3.1 Data used in this work

To validate the method we restrict this analysis to a mock galaxy catalogue corresponding to a single snapshot at $z = 0.57$. In particular, it matches the CMASS sample of luminous red galaxies (LRGs), which is a complete sample, nearly constant in mass and volume, limited between the redshifts $0.43 \leq z \leq 0.7$ (see Anderson et al. (2014) for details of the targeting strategy). We use the N -body based mock galaxy catalogue constructed to match the clustering bias and number densities of the BOSS DR12 CMASS galaxies at the mean redshift of $\bar{z} = 0.57$.

The mock galaxy catalogue used in this study was presented in Rodríguez-Torres et al. (2016) and was extracted from the BigMDPL N -body simulation,² one of the Multidark simulation project, which was performed using the GADGET-2 code (Springel 2005). The BigMDPL was run with 3.840^3 particles on a volume of $(2.5 h^{-1} \text{Gpc})^3$ assuming Λ CDM Planck cosmology with $\{\Omega_\Lambda = 0.6928, \Omega_M = 0.307, \Omega_b = 0.0482, \sigma_8 = 0.828, n_s = 0.961\}$, and a Hubble constant ($H_0 = 100 h \text{ km s}^{-1} \text{ Mpc}^{-1}$) given by $h = 0.677$. Halos and subhalos were identified using the ROCKSTAR halo finder (Behroozi, Wechsler & Wu 2013).

3.2 Results

For our study we rely on the COSMIC BIRTH code (Kitaura et al. 2020) to sample the density field with the lognormal-Poisson model, switching off: displacements, peculiar motions, and selection effects. This corresponds to the first Gibbs-sampling step listed in the COSMIC BIRTH paper, which initially represented the bottle-neck of the computations. The posterior distribution function is sampled with the HMC sampling technique following the methods described in the previous section, including an automatic estimation of the logarithmic mean field μ (see Kitaura, Gallerani & Ferrara 2012a).

We choose first two meshes of 128^3 and of 256^3 on a cubical volume of $1250 h^{-1} \text{Mpc}$ side. We perform a nearest-grid-point mass assignment of the mock galaxy catalogue on the grid to obtain the data array, as the number counts per cell.

At the final part of our analysis, we consider also meshes of 64^3 to verify how second-order schemes start becoming more efficient towards lower statistical dimensions.

3.2.1 Parallelization and optimal number of cores

The numerical tests have been performed using the *Diva Severo Ochoa* machine, which is a High Performance Computer at the IAC with specifications shown in Table 1.

First, a study of the optimal number of cores to run the Open-MP parallel COSMIC BIRTH code is presented. To do so, the code has been run for the same parameters (i , step size ϵ , seed and number of iterations) for different number of cores: 1, 2, 4, 8, 16, 32, and 64.

(i) Low-resolution case: 128^3 cells.

The left-hand panel in Fig. 1 shows the computational time needed to reach 100 iterations as a function of the number of cores, represented by the red line. The black line is the reference one from a perfect scaling of the computation time with the number of cores, which means that the computational time decreases to the half each time we double the number of cores. As we can see, for more than 8 cores, the computation time decreases slowly until it becomes almost constant for more than 32 cores. Hence, the computation time saved using 16, 32, or 64 cores is not remarkable enough compared to using 8, as it deviates from the ideal case (black line). For this reason all the runs with 128^3 cells in this study were performed with 8 cores.

(ii) High-resolution case: 256^3 cells. The right-hand panel in Fig. 1 represents the speed up factor by the solid red line, which is defined by the largest time of all runs (the one for 1 core) divided by the time of each run. The solid black line shows the reference curve for an ideal speed up factor: 1 for 1 core, 2 for 2 cores, and so on. In this case, until 32 cores, we find that the speed up factor goes approximately as the ideal case. However, for 64 cores we can see that there is a deviation with respect to the solid black line. For this reason, we choose 32 cores for the high-resolution in this study. We have chosen a different representation here as for the low-resolution case, to better assess the saturation for large number of cores.

We note, however, that our study is not affected by the chosen architecture, as we will express the efficiency as a function of the number of evaluations of the Hamiltonian equations of motion,

²See <https://www.cosmosim.org/cms/simulations/bigmdpl/>

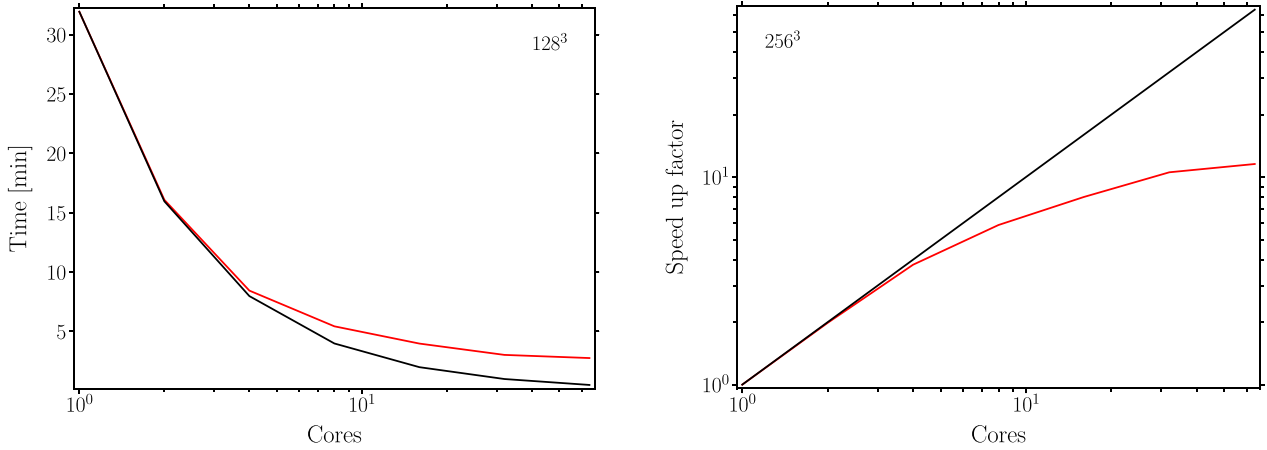


Figure 1. The left-hand panel (used for the 128^3 case) shows the computation time as a function of the number of cores (solid red line). The solid black line represents the reference ideal case, in which the computation time decreases to the half each time we double the number of cores. The right-hand panel (used for the 256^3 case) shows the speed up factor as a function of the number of cores. The solid black line represents the ideal case, in which the speed up factor is 1 for 1 core, 2 for 2 cores, and so on.

which is directly related to the number of gradient computations (see Section 2.2).

3.2.2 Convergence criteria

To determine the iteration at which the HMC sampler reaches convergence, we compare the power spectrum of a specific iteration with a reference converged power spectrum. The latter is obtained with the second-order leap-frog algorithm, from computing the average power spectrum from iteration 3000 to 12 000, i.e. taking samples well after the chain has passed the burn-in phase. We estimate that convergence has set in when the ratios between the reference power spectrum and that of a certain iteration are compatible with each other within 2.5 per cent. We further assess the convergence of the chains in a robust way using the Gelman-Rubin estimator in Section 3.2.4.

3.2.3 Parameter study: optimal step size and convergence

To define our reference computation we start with the second-order leap-frog algorithm, and determine an optimal step size of $\epsilon = 0.06$, multiplied by a uniform random number with an additionally drawn random number of steps in each iteration, N_{eval} , shown in equation (33) (see Neal 1993, and Section 2.2.2).

For the fourth-order discretization scheme, the optimal setup will be investigated in the following subsections. We start with the low-resolution case, which permits us to scan more broadly the parameter space.

(i) Low-resolution studies: 128^3 cells

This study has been done for different number of steps i within one iteration, and for different step sizes (multiple values of ϵ), to analyse the convergence, the computation time, and the acceptance rate. This last parameter expresses the percentage of iterations that have been accepted at the first time. A too large step size will result in a very low acceptance rate for the new states, and a too small step size can waste computation time or will lead to a slow exploration of the parameter space.

Table 2 shows the iteration at which the chain converges, the number of evaluations of the Hamilton's equations of motion required to achieve that convergence, and the acceptance rate. This has been

Table 2. Convergence as a function of the step-size value, iteration, number of evaluations (NoE) of the Hamiltonian equations of motion, and acceptance rate, obtained with the fourth-order leap-frog algorithm for the low-resolution (128^3) runs. The global step size is given by $\Delta\tau_{4\text{th}} = (2i - (2i)^{1/3})m\epsilon$ for $m = 1, 2, 4, 6, 8, 10$, with $\epsilon = 0.06$.

Step size $m\epsilon$	Iteration of convergence	CONV [NoE]	Acceptance (per cent)
<i>i</i> = 1			
0.5 ϵ	2530	7722	98.27
ϵ	650	2012	97.66
2 ϵ	250	916	83.51
4 ϵ	230	1544	51.03
6 ϵ	250	2460	33.00
8 ϵ	258	3605	24.05
10 ϵ	232	4003	19.09
<i>i</i> = 2			
0.5 ϵ	245	1246	98.46
ϵ	68	366	93.30
2 ϵ	53	533	52.47
4 ϵ	42	740	29.92
6 ϵ	46	1136	19.92
8 ϵ	46	1660	16.93
10 ϵ	36	1892	9.08
<i>i</i> = 3			
0.5 ϵ	209	1485	98.13
ϵ	32	360	76.58
2 ϵ	29	582	43.42
4 ϵ	20	638	23.01
6 ϵ	24	1196	13.54
8 ϵ	25	1672	13.96
10 ϵ	27	1470	11.70
<i>i</i> = 4			
0.5 ϵ	100	929	96.53
ϵ	39	486	81.76
2 ϵ	32	785	45.60
4 ϵ	27	981	26.22
6 ϵ	26	1634	15.04
8 ϵ	27	2342	13.66
10 ϵ	26	2268	6.66

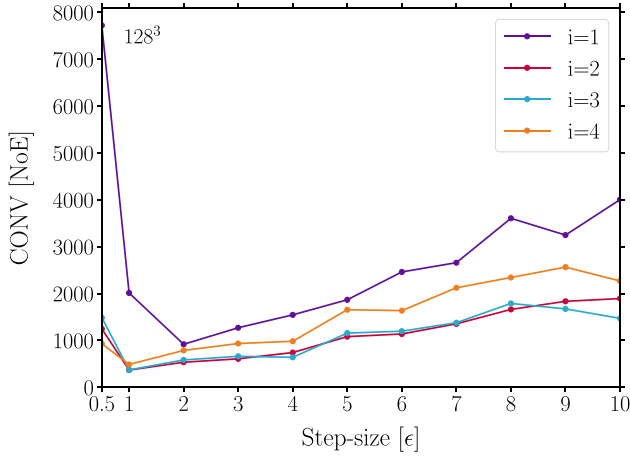


Figure 2. Number of evaluations of the Hamiltonian equations of motion required to achieve the convergence as a function of the step size for the different values of i .

computed for each value of i and different step sizes. To suppress the dependence on the starting point of the chain, all runs have been performed for five different seeds, yielding stable results, as can be seen in the small fluctuations in Fig. 2. Hence, the results in Table 2 represent the average over the five chains. The highest acceptance rates are obtained for a step size of 0.5ϵ in all four cases of i . However, the convergence is achieved at a higher iteration than for the other configurations. For a step size of ϵ the acceptance rate is still high, and the iteration of convergence has significantly decreased with respect to the previous case. Thus, for the cases of $i = 2, i = 3$ and $i = 4$ this is the optimal configuration. For the case of $i = 1$, Table 2 shows that the optimal step size value is the one of 2ϵ , for which the convergence is reached at iteration 250 with a relative high acceptance rate, while for a step size of ϵ we need 650 iterations to converge. However, as we increase the step-size value, we can observe that convergence is achieved at a similar number of iterations as in the case of 2ϵ , but with at the expense of a higher computational cost due to the

number of rejected samples. We also find that, for larger values of i , the acceptance ratio decreases faster with increasing step sizes, with some exceptions for $i = 4$, which implies that the computation time increases. This can also be seen in Fig. 2, where we have represented the convergence time over the step-size value. In particular, we find a linear positive slope from a step size of ϵ to 10ϵ , with the exception of step size ϵ for case $i = 1$. In this case, the global transformation to a new state, including the backward step, is presumably too short to take advantage of the fourth-order discretization (see discussion at the end of Section 2.2.3). This effect can also be observed taking a step size of 0.5ϵ , especially for $i = 1$, which implies a significant increase of the number of evaluations of the Hamiltonian equations of motion and, therefore, a higher computational cost.

Fig. 3 represents the acceptance for the case $i = 1$ as a function of the step size $m\epsilon$. We can see that, as the step size increases, the number of rejections becomes larger. For the case of a step size of ϵ , 97.0 per cent of the iterations are accepted at the first trial, and 3.0 per cent at the second one, i.e. with only one rejection. For the case of 2ϵ , we can observe that there is increment, although small number of iterations that are accepted at the second and third time. The histogram in the lower right-hand panel of Fig. 3, for a step-size value of 10ϵ , shows that iterations can be rejected up to 50 times before being accepted, which dramatically increases the computational cost. The same behaviour has been found for cases $i = 2, i = 3$ and $i = 4$ as it is shown in Table 2.

Fig. 4 shows the acceptance for a step size of ϵ for the four studied values of i . We can see in the panel on the left that, for the case of $i = 1$, almost all iterations are accepted without rejections. For $i = 2$ we find that there is a very high acceptance ratio at the first trial, but some rejections start to appear after one trial. For $i = 3$ the number of rejections increase, although remaining low. Finally, for $i = 4$ we find, in the panel of the right, a similar behaviour to the previous case, decreasing the number of rejections at high trials.

(ii) High-resolution studies: 256^3 cells.

Once we have studied the low-resolution case, we can now focus on fewer configurations at a higher resolution. For the second-order leap-frog algorithm we investigate, first, the optimal N_{eval} which goes into equation (33). Previous studies at lower resolution (128^3 cells)

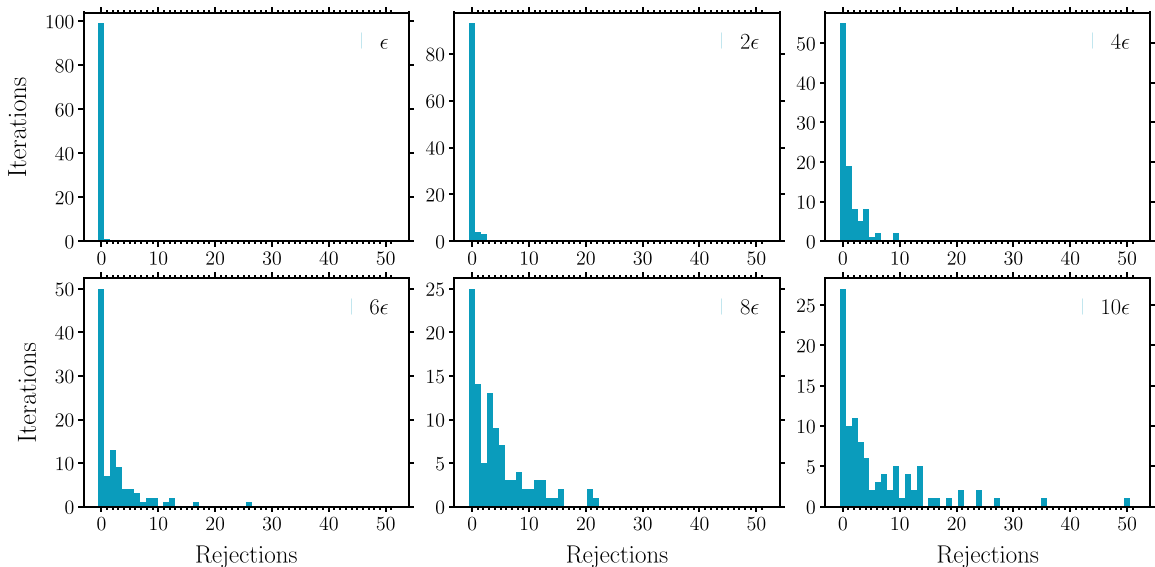


Figure 3. Number of rejected iterations for $i = 1$ and the different values of step size for the 128^3 runs in the fourth-order scheme.

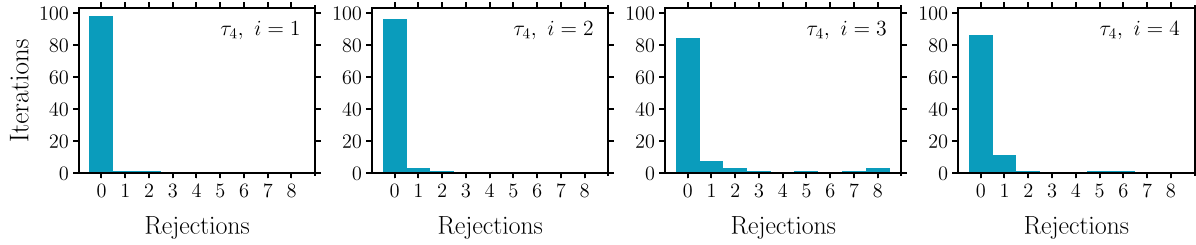


Figure 4. Number of rejected iterations for a step size of ϵ and the different values of i in the fourth-order scheme. From the left to the right: $i = 1, 2, 3$, and 4 for the 128^3 runs.

showed an optimal step-size value of ϵ for different configurations, i.e. different N_{eval} values. Hence, we have chosen this step size in all the cases of this scheme shown in Table 3. In the fourth-order method we present the results for the configurations 2ϵ , for $i = 1$ and ϵ for $i = 2, i = 3$ and $i = 4$. These were the most efficient step sizes for each value of i , as we could see in Table 2 and Fig. 2. In Table 3, we can compare the second and fourth-order leap-frog schemes at a higher resolution of 256^3 cells. Each value of the table

is an average over 4 different seeds. We have empirically found that for our setting this is a reasonable number to avoid being much affected by the initial conditions. In fact, in a number of relevant cases, several measures have not changed, such as the number of evaluations or the correlation length for the fourth-order $i = 3$ case (see Section 3.2.5). The table shows seven N_{eval} values from 2 to 100 in the second-order case, and six configurations for the fourth-order method: four different i values from 1 to 4, and the last two

Table 3. Comparison between the second and fourth-order leap-frog schemes (\mathcal{T}_2 and \mathcal{T}_4 , respectively) for the high-resolution (256^3) runs. The first column indicates the specific settings of the scheme, the second column shows the average number of evaluations of Hamilton’s equations of motion (NoE), the third column shows the iteration of convergence, and the fourth column shows the number of evaluations until convergence (taking into account the rejections). In the fifth column we have the improvement factor (IF) of each scheme versus the fastest \mathcal{T}_2 in convergence (CONV) and the 6th column shows the acceptance rate (AR_{CONV}) until convergence. The \mathcal{T}_4 with $i = 1$ was run with a basic step size of 2ϵ , while the rest used ϵ instead, following the analysis shown in Fig. 2. This results in global step sizes given by equations (35) and (47) indicated in the seventh column. The last column shows the effective global step size weighted with the respective AR_{CONV} (equation 48). The $N_{\text{eval}} = 2$ case is the one with the least number of evaluations, which converges. One might be cautious about this case, given the risk of resonant trajectories for such a low number of evaluations ($\langle \text{NoE} \rangle = 1.5$). In fact, the $N_{\text{eval}} = 1$ case does not converge in general. The best second- and fourth-order cases are highlighted in grey. We choose the best fourth-order case with a fixed number of evaluations as the reference (ref).

Scheme	(NoE)	CONV [NoI]	CONV ^{eff} [NoE]	IF _{CONV}	AR _{CONV} (per cent)	$\Delta\tau_{\text{CONV}}$ [ϵ]	$\Delta\tau_{\text{CONV}}^{\text{eff}}$
Second order							
\mathcal{T}_2							
$N_{\text{eval}} = 2$	1.5	4320	12557	28.22	61.11	0.68	0.277
\mathcal{T}_2							
$N_{\text{eval}} = 5$	3	2175	14833	33.33	46.96	0.83	0.130
\mathcal{T}_2							
$N_{\text{eval}} = 15$	8	665	17372	39.04	33.83	1.34	0.057
\mathcal{T}_2							
$N_{\text{eval}} = 30$	15.5	405	27459	61.71	23.78	3.87	0.059
\mathcal{T}_2							
$N_{\text{eval}} = 50$	25.5	242	33069	74.31	20.82	4.85	0.039
\mathcal{T}_2							
$N_{\text{eval}} = 80$	40.5	149	45738	102.78	15.63	7.25	0.028
\mathcal{T}_2							
$N_{\text{eval}} = 100$	50.5	113	53789	120.87	15.46	7.54	0.023
Fourth order							
\mathcal{T}_4							
$i = 1$	3	340	1496	3.36	69.85	1.48	0.35
\mathcal{T}_4							
$i = 2$	5	100	608	1.37	83.79	2.41	0.40
\mathcal{T}_4							
$i = 3$	7	33	445	ref	71.30	4.18	0.43
\mathcal{T}_4							
$i = 4$	9	33	504	1.13	67.13	6.00	0.45
Fourth order with random i							
\mathcal{T}_4							
$i \in \{1, 2, 3, 4\}$	6	41	349	0.78	75.54	3.27	0.42
\mathcal{T}_4							
$i \in \{2, 3, 4, 5, 6\}$	9	25	439	0.99	60.03	5.22	0.35

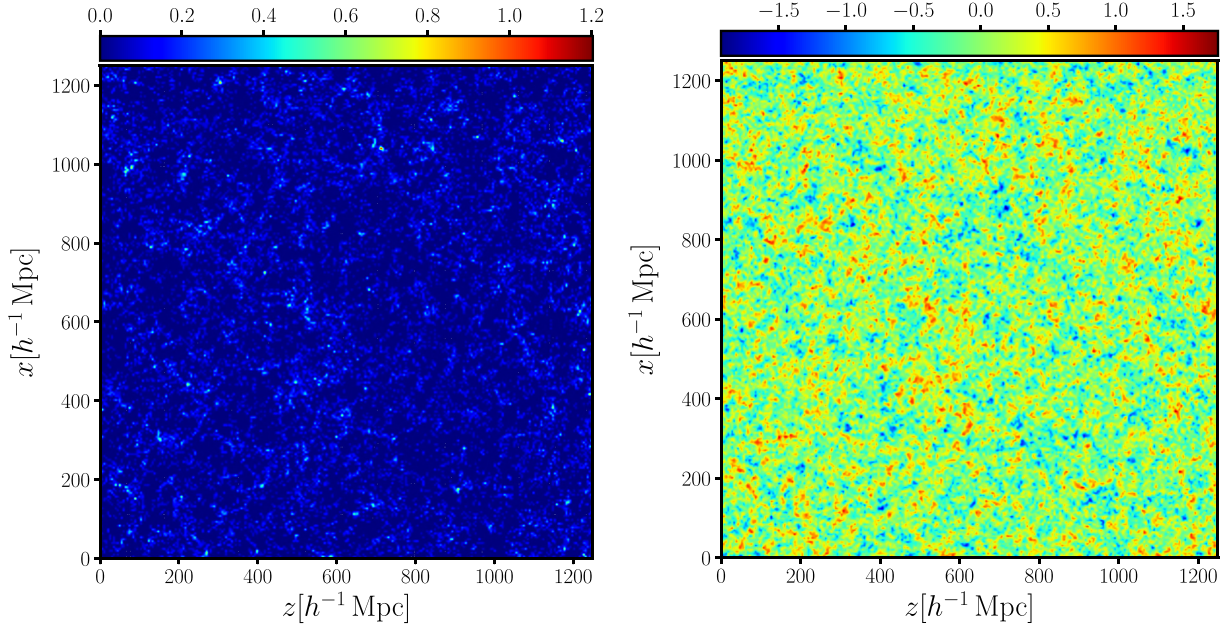


Figure 5. Comparison between the halo density field (left) and the reconstructed primordial fluctuations, $\delta(x)$, with fourth-order leap-frog algorithm, for $i = 3$ and a step size $\epsilon = 0.06$ (right). We have taken a volume of $(1250 h^{-1} \text{ Mpc})^3$ and 256^3 cells. The slice was obtained integrating 10 cells in the y -direction, corresponding to a thickness of $\sim 50 h^{-1} \text{ Mpc}$.

cases of the table, with a random value of i in each iteration, from 1 to 4 and from 2 to 6, respectively. Table 3 shows the average number of evaluations of Hamilton's equations of motion required in each iteration ($\langle \text{NoE} \rangle$), the iteration of convergence (NoI) and the number of evaluations until that convergence, taking into account the rejections (NoE). From this study we obtain the ratio of the number of evaluations until convergence is reached between each case and that of the reference (IF). We can also find the acceptance rate until the convergence (AR) and the global step sizes given by equations (41) and (47), with the effective global step size weighted with respect to the acceptance rate as follows:

$$\Delta \tau^{\text{eff}} \equiv \frac{\Delta \tau \cdot \text{AR}}{\langle \text{NoE} \rangle}. \quad (48)$$

This quantity is defined based on the time step size $\Delta \tau$, penalized by the acceptance rate AR, and the number of evaluations (NoE).

We have chosen as the fourth-order reference case, the one of $i = 3$, which is the most efficient one in terms of convergence from the cases $i = 1, 2, 3$ and 4. We note that the fourth-order case has a random step size, following equation (47), but the number of evaluations is in general fixed. For this reason, we tested also the behaviour of the fourth-order scheme with random number of evaluations (randomizing i) as shown in the last two rows of Table 3. The convergence and correlation length (shown in Section 3.2.5) are not significantly better to that one of $i = 3$. Thus, moderate improvements could be achieved, by considering more cases, than the ones computed in this study. For the second-order method, represented in the seven first rows of the table, we find an optimal configuration in terms of convergence for the lowest number of evaluations of $N_{\text{eval}} = 2-5$. This is however, a factor ~ 30 less efficient than the best higher fourth-order case. The $N_{\text{eval}} = 2$ case with $\langle \text{NoE} \rangle = 1.5$ has converged considering different seeds, however, for $N_{\text{eval}} = 1$, we did not find stable convergence, as there is a high risk of producing resonant trajectories.

A visual impression of the reconstruction is shown in Fig. 5, where the input catalogue and the corresponding reconstruction of the linear density field using the fourth-order discretization scheme are shown. Here we can qualitatively verify that the discrete number counts of objects on the left-hand panel is translated into a continuous density field on the right-hand panel. This is essential to primordial density reconstructions, as we need to obtain a clean Gaussian field on which we can make nonlinear cosmic evolution operations (see e.g. scheme in Kitaura 2013, relating the Gaussian field to the final galaxy distribution, and the corresponding power spectra).

Fig. 6 shows the convergence of the power spectra as a function of the number of iterations for the optimal case of the second-order leap-frog algorithm ($\mathcal{T}_2, N_{\text{eval}} = 5$) and for the cases of $i = 1, 2$, and 3 of the fourth-order scheme. While in the second-order method convergence is reached at iteration ~ 2200 , with the fourth-order scheme we get converged samples at iteration ~ 280 for the case $i = 1$, ~ 100 for $i = 2$, and ~ 40 for $i = 3$, as we can read from Table 3. Note that Fig. 6 shows results for a particular seed, so that the optimal values can vary with respect to the ones presented in the table, as the average over different seeds. The ratio between the converged reference power spectrum, represented with the black line, and the power spectrum at the estimated convergence iteration, are shown in the lower panels of Fig. 6. We find that the ratios are compatible within a 2.5 per cent error, represented with the red band.

To further assess the convergence of the second and fourth-order leap-frog algorithms, we have performed an additional analysis. In particular, we compute, for each iteration, the difference of the individual power spectrum and the reference one, summing up for all modes, k , as represented in Fig. 7. Convergence is achieved when this difference is smaller than a threshold (in this case, the criterion is < 1) for at least 10 consecutive iterations. As this is for one seed, we can see the convergence iterations are in agreement with Fig. 6 and Table 3. It is also remarkable to observe that the modes averaged difference between the reference and sampled power spectra at

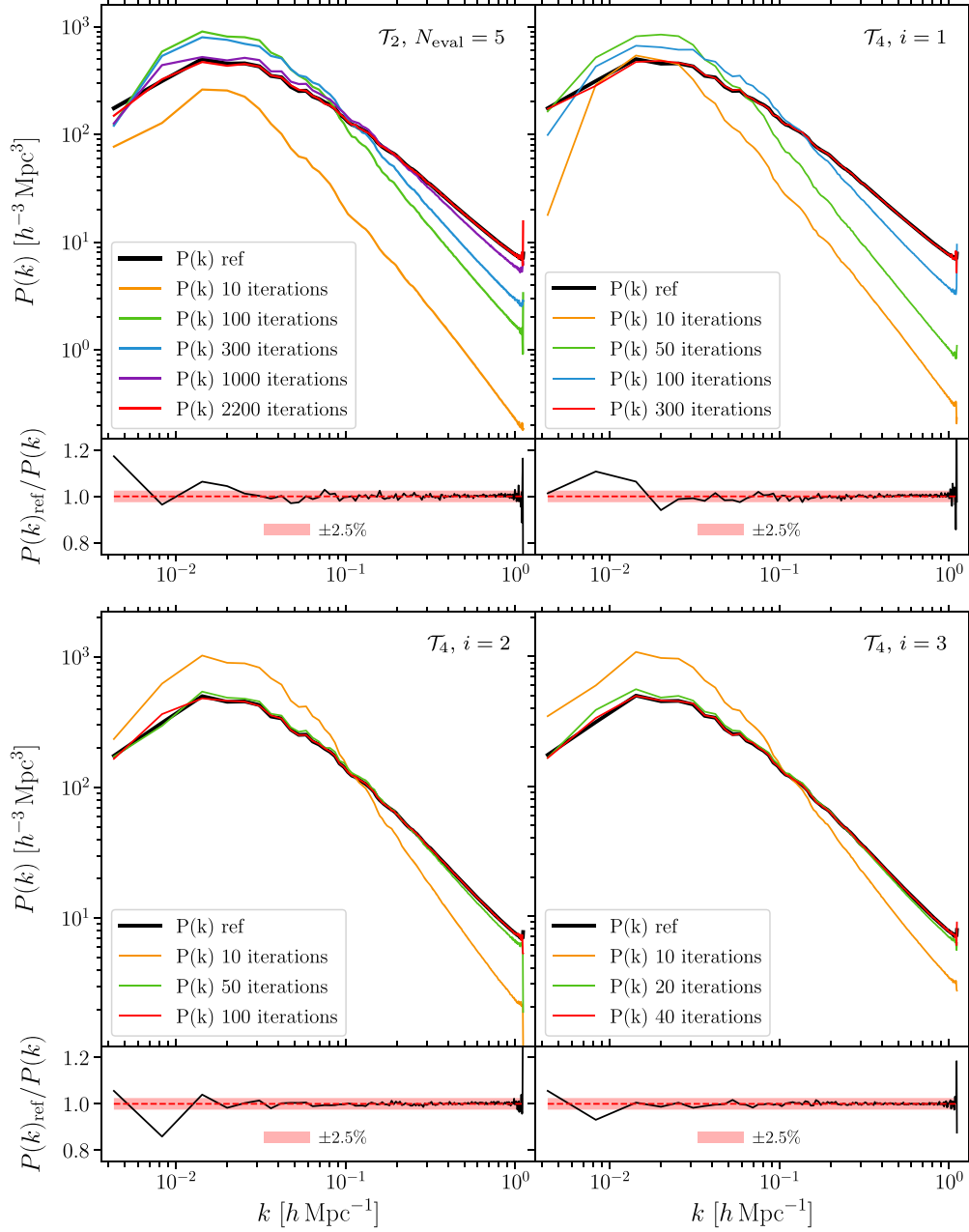


Figure 6. Power spectra for different iterations (coloured lines) compared to the reference averaged (over 9000 samples) converged power spectrum (black line) for the high-resolution (256^3) runs. The lower panels show the ratio between the converged sample for each set-up with respect to the reference converged sample. The power spectrum represented with the red line corresponds to that at iteration of convergence. The subplots show the ratio between the reference converged power spectrum and the converged one for each setting. Upper left: power spectrum for different iterations with the second-order leap-frog algorithm. Upper right: power spectrum for different iterations with the fourth-order leap-frog algorithm, for $i = 1$ and a step size 2ϵ . Lower left: power spectrum for different iterations with the fourth-order leap-frog algorithm, for $i = 2$ and a step size ϵ . Lower right: power spectrum for different iterations with the fourth-order leap-frog algorithm, for $i = 3$ and a step size ϵ .

different iterations is always considerably closer to zero for the fourth-order case than for the second-order one. Then, we represent the ratio of the reference power spectrum with the power spectrum at that iteration (previously estimated), and the next ones, where we find that they are compatible with each other within 2 per cent. This test is reassuring, as both convergence criteria are in excellent agreement.

Having set the number of forward steps to $i = 3$, and the step size to ϵ , which is the most optimal configuration, we can now proceed

to study the convergence of the fourth-order leap-frog algorithm compared to the second-order one in an additional robust way.

3.2.4 Robust convergence assessment: Gelman-Rubin test

To verify that convergence has been reached at iteration ~ 30 , we perform the Gelman-Rubin test. Multiple chains are supposed to converge to some stationary distribution. Hence, comparing the

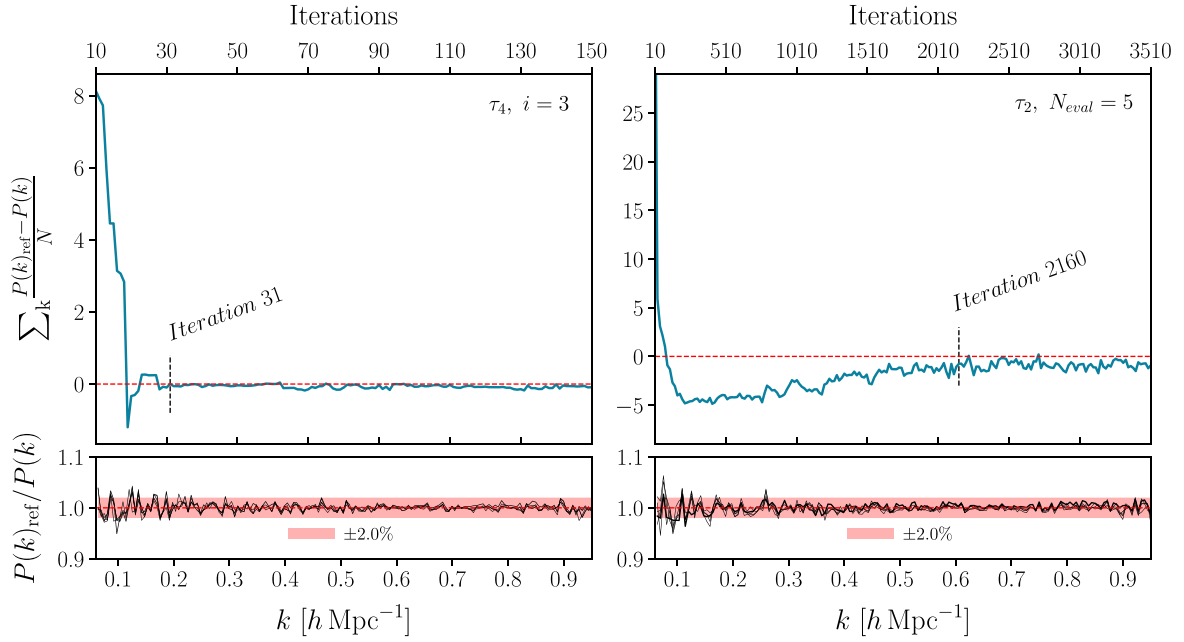


Figure 7. Difference of the power spectrum of each iteration and the reference one, summing up for all modes, $0.06 \, h \, \text{Mpc}^{-1} < k < 0.95 \, h \, \text{Mpc}^{-1}$ to avoid cosmic variance. The left-hand panel shows the fourth-order scheme, $\tau_4, i = 3$, and the right-hand panel shows the second-order scheme, $\tau_2, N_{\text{eval}} = 5$. The lower panels show the ratio of the converged power spectrum and the consecutive ones, with the black lines.

mean and variance within one converged chain to the samples of independent chains, gives a tool to verify convergence of Markov chains. In this test we have to run N_{chains} of length N_{length} , that are supposed to have the same target distribution, but starting at different points, so each one has a different seed. The output of the chain is represented by $x_{c,s}$, with $c \in 1, 2, \dots, N_{\text{chains}}$ and $s \in 1, 2, \dots, N_{\text{length}}$. x is, in this case, the over-density δ_i of each cell. The goal is to compare the variance of the N_{chain} means of the different chains to the mean of the variance of each individual chain. The parameter R introduced in Gelman & Rubin (1992), known as the Potential Scale Reduction Factor (PSRF), is assumed to represent a converged chain when reaching a value of $R = 1.1$.

We first calculate each chain's mean value

$$\bar{x}_c = \frac{1}{N_{\text{length}}} \sum_s x_{c,s}. \quad (49)$$

Then we calculate each chain's variance

$$\sigma_c^2 = \frac{1}{N_{\text{chains}} - 1} \sum_s (x_{c,s} - \bar{x}_c)^2. \quad (50)$$

Then, we determine all chain's mean

$$\bar{x} = \frac{1}{N_{\text{chains}}} \sum_c \frac{1}{N_{\text{length}}} \sum_s x_{c,s} = \frac{1}{N_{\text{chains}}} \sum_c \bar{x}_c. \quad (51)$$

The weighted mean of each chain's variance is expressed as

$$B = \frac{N_{\text{length}}}{N_{\text{chains}} - 1} \sum_c (\bar{x}_c - \bar{x})^2, \quad (52)$$

and the average variance by

$$W = \frac{1}{N_{\text{chains}}} \sum_c \sigma_c^2. \quad (53)$$

Finally, the PSRF is defined as

$$R = \sqrt{\frac{N_{\text{length}} - 1}{N_{\text{length}}} + \frac{N_{\text{chains}} + 1}{N_{\text{length}} N_{\text{chains}}} \frac{B}{W}}. \quad (54)$$

We have represented the range in which the Markov chain has converged and, therefore, where the HMC has reached the target distribution. As it is mentioned before, we evolve the system with Hamilton's equations of motion. However, the initial samples do not belong to the correct target distribution, but are part of the *burn-in* phase. Fig. 8 presents the results of the Gelman-Rubin test for the fourth-order leap-frog algorithm, as compared to the second-order one. This calculation has been done for four different chains. The upper panel of Fig. 8, shows that a small range of 40–500 iterations, already gets the majority of the points below the solid red line, which represents $R - 1 = 0.1$. However, for second-order leap-frog algorithm we need a larger range to find a similar behaviour in the Gelman-Rubin test: from 3000 to 12 000 (lower panel of Fig. 8). If we take the same range as for the fourth-order one, we can verify that the Markov chain is far from converged (see middle panel in Fig. 8).

3.2.5 Correlation length

Finally, we compute the correlation length of the density bins over the iteration distance. The correlation length is calculated as

$$C_n(\sigma_j) = \frac{1}{N - n} \sum_{i=0}^{N-n} \frac{(\delta_j^i - \langle \delta_j \rangle) (\delta_j^{i+n} - \langle \delta_j \rangle)}{\sigma^2(\delta_j)}, \quad (55)$$

where δ_j is the overdensity field in each iteration, N is the number of samples and n is the distance between iterations. Some particular computations are shown in Fig. 9.

Table 4 shows the correlation length for different configurations previously studied (summarized in Table 3). In particular, it shows

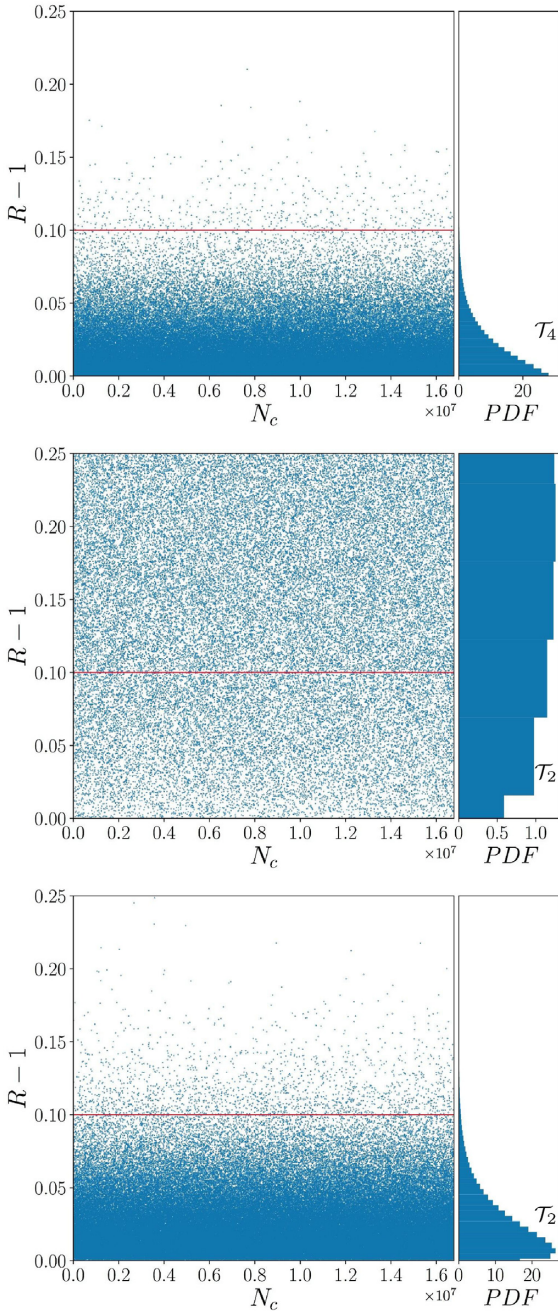


Figure 8. Upper panel: Gelman–Rubin test from 40 to 500 iterations with the best fourth-order leap-frog algorithm, for the configuration $i = 3$. Middle panel: Gelman–Rubin test from 3000 to 3460 iterations with second-order leap-frog algorithm with $\langle \text{NoE} \rangle = 5.5$. Lower panel: Gelman–Rubin test from 3000 to 12 000 iterations for second-order leap-frog algorithm with $\langle \text{NoE} \rangle = 5.5$. The red line represents the $R - 1 = 0.1$ parameter.

the correlation length in terms of evaluations of the Hamiltonian equations of motion and the improvement factor of each case in producing independent samples, given by the ratio between these evaluations and those ones of the reference scheme (\mathcal{T}_4 , $i = 3$). We also find the number of independent samples produced with \mathcal{T}_4 , $i = 3$ until each scheme converges

$$\text{NoIS} \equiv \frac{\text{CONV}^{\text{eff}}[\text{scheme}] - \text{CONV}^{\text{eff}}[\tau_{4i=3}]}{\text{CL}^{\text{eff}}[\tau_{4i=3}]} \quad (56)$$

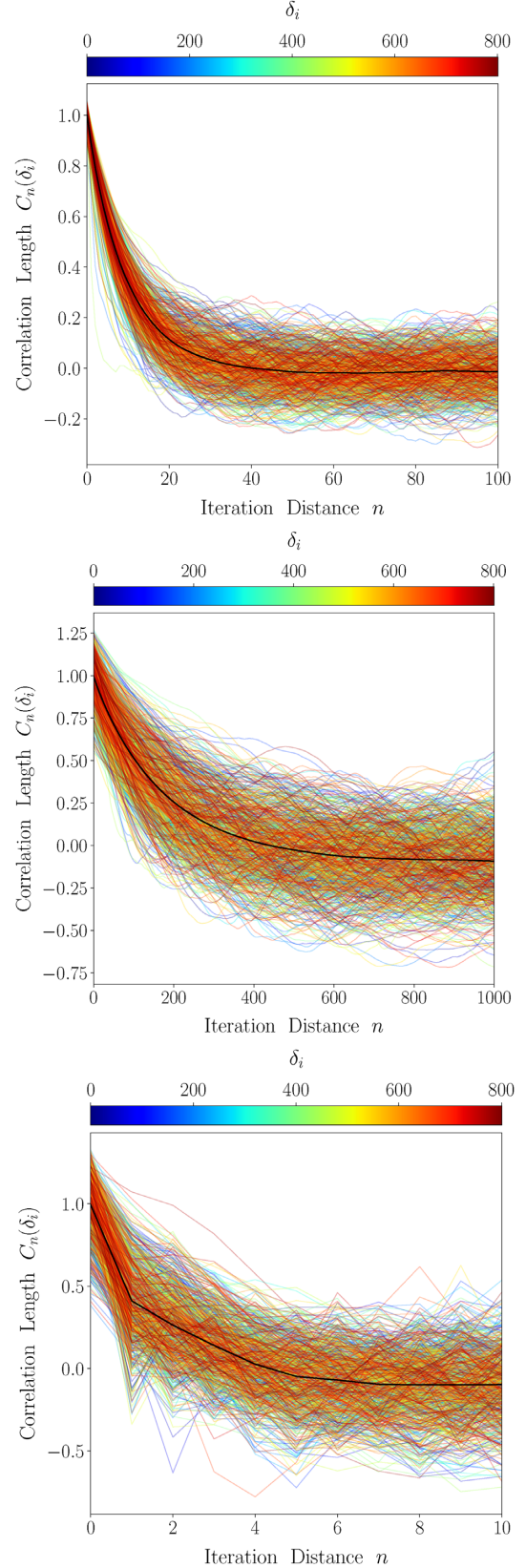


Figure 9. Correlation length for 800 randomly chosen density voxels δ_i as a function of the iteration distance. Upper panel: \mathcal{T}_4 , $i = 3$, middle panel: \mathcal{T}_2 , $N_{\text{eval}} = 5$, and lower panel: \mathcal{T}_2 , $N_{\text{eval}} = 80$. The black solid line represents the mean over all density fields.

Table 4. Comparison between the second- and fourth-order leap-frog schemes (\mathcal{T}_2 and \mathcal{T}_4 , respectively) for the high-resolution (256^3) runs. The first column indicates the specific settings of the scheme, the second column shows the average number of evaluations of Hamilton's equations of motion (NoE), the third column shows the correlation length in terms of iterations (NoI), and the fourth column shows the effective correlation length (CL^{eff}) in units of NoE (taking into account the rejections). The fifth column indicates the improvement factor of each scheme in producing independent samples after convergence versus the $\mathcal{T}_4, i = 3$ configuration. The sixth column shows the acceptance rate (AR) after the convergence and the seventh column indicates the number of independent samples (NoIS) produced with $\mathcal{T}_4, i = 3$ until each scheme converges. \mathcal{T}_4 with $i = 1$ was run with a basic step size of 2ϵ , while the rest used ϵ instead, following the analysis shown in Fig. 2. This results in global step sizes given by equations (35) and (47) indicated in the eighth column. The last column shows the effective global step size weighted with the respective AR, given by equation (48). The last column shows the effective time steps $\Delta\tau^{\text{eff}}$. The best second- and fourth-order cases are highlighted in grey. We choose the best fourth-order case with a fixed number of evaluations as the reference (ref).

Scheme	(NoE)	CL [NoI]	CL^{eff}	IF _{CL}	AR _{CL} [%]	NoIS	$\Delta\tau_{\text{CL}}$ [ϵ]	$\Delta\tau_{\text{CL}}^{\text{eff}}$
2nd order								
\mathcal{T}_2								
$N_{\text{eval}} = 2$	1.5	290	527	3.66	80.29	75	0.76	0.407
\mathcal{T}_2								
$N_{\text{eval}} = 5$	3	200	826	5.13	74.33	89	1.49	0.369
\mathcal{T}_2								
$N_{\text{eval}} = 15$	8	70	904	5.61	60.37	105	4.00	0.302
\mathcal{T}_2								
$N_{\text{eval}} = 30$	15.5	22	609	3.78	60.39	167	7.69	0.300
\mathcal{T}_2								
$N_{\text{eval}} = 50$	25.5	9	433	2.69	59.09	203	12.42	0.288
\mathcal{T}_2								
$N_{\text{eval}} = 80$	40.5	4	369	2.29	55.38	281	19.61	0.268
\mathcal{T}_2								
$N_{\text{eval}} = 100$	50.5	4	390	2.42	57.40	331	24.23	0.227
4th order								
\mathcal{T}_4								
$i = 1$	3	123	2000	12.42	41.81	7	1.48	0.206
\mathcal{T}_4								
$i = 2$	5	55	474	2.94	83.20	1	2.41	0.401
\mathcal{T}_4								
$i = 3$	7	18	161	ref	83.53	ref	4.18	0.498
\mathcal{T}_4								
$i = 4$	9	16	163	1.01	84.5	0.4	6.12	0.575
4th order with random i								
\mathcal{T}_4								
$i \in \{1, 2, 3, 4\}$	6	28	214	1.33	82.48	-0.6	3.32	0.456
\mathcal{T}_4								
$i \in \{2, 3, 4, 5, 6\}$	9	14	263	1.63	73.54	-0.04	5.43	0.443

Then, we have the acceptance rate after the convergence, and the global step size, also in this range, including the effective one. We can conclude from Table 4 that, for the second-order leap-frog algorithm, the optimal configuration is the one of $N_{\text{eval}} = 80$. However, although this case has a very low correlation length of four iterations, due to the high value of evaluations of the Hamilton's equations of motion in each iteration (we have an average of 40.5 as we can see in the second column of the table), the fourth-order method is still more efficient, as we discuss in detail below, having an average correlation length of 18 iterations. Another important aspect to consider here is that, while the case $\mathcal{T}_2, N_{\text{eval}} = 80$ converges, the $\mathcal{T}_4, i = 3$ has already produced ~ 280 independent samples. We can also see that, although for the convergence the case of $\mathcal{T}_2, N_{\text{eval}} = 5$ was closely the optimal one, for the correlation length it is a factor ~ 5 times worse than the fourth-order method.

Fig. 9 shows the correlation length of for the different leap-frog algorithms. The upper panel presents the best fourth-order case, while the middle and lower panels show the second-order cases for $N_{\text{eval}} =$

5 and $N_{\text{eval}} = 80$, respectively. The black solid line represents the mean over all density voxels δ_i , where we assume that we have independent samples, when the correlation length is lower than 0.1. We have defined an effective time step size shown in equation (48) trying to understand the trends seen in Tables 3 and 4. All fourth-order schemes show larger effective time steps than the second-order ones. We also find that the methods with the largest effective time steps converge faster (see Table 3).

However, the shorter correlation length for the second-order case, achieved at $N_{\text{eval}} = 80$, has not the largest effective time step size (table 4).

3.2.6 Effective correlation length

Larger integration times resulting from the product of the time step size and the number of evaluations will produce less correlated samples. However, there is a trade-off from increasing the number of

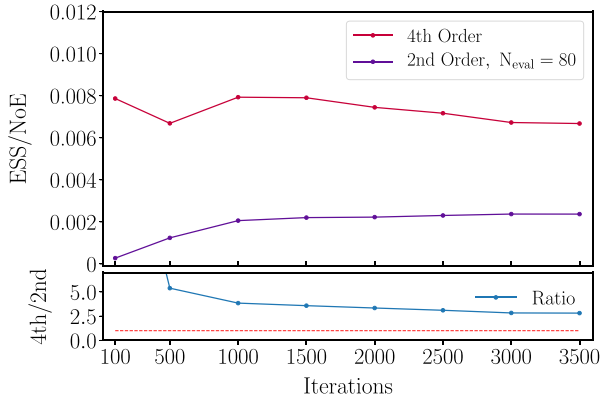


Figure 10. ESS estimator divided by the number of evaluations of the Hamilton's equations of motion (NoE). The lower caption shows the ratio between the fourth- and the second-order cases. The fourth-order method is on average a factor 3.2 superior to second-order one, considering the range after 1000 iterations (2.8 for the range 3000–3500 iterations). We have checked that the rest of N_{eval} cases (including $N_{\text{eval}} = 50$) perform worse than $N_{\text{eval}} = 80$.

evaluations, and at some point, the high $\langle \text{NoE} \rangle$ combined with low acceptance rates do not compensate for the shorter CL [NoI]. Also, towards larger integration time steps, the acceptance rate diminishes. What counts at the end is the effective correlation length CL^{eff} .

One needs to ask how many gradient evaluations NoEs are required to obtain the number of accepted iterations given by CL[NoI]. Across a chain after convergence, this effective CL fluctuates depending on the rejections, which have to be included in the computation of the NoEs. Therefore, we take the average. Hence, the effective correlation length CL^{eff} is computed as the average number of evaluations (NoEs) including rejected samples, required to obtain the number of accepted iterations as indicated by the respective correlation length CL. The average is computed well after convergence, taking several thousands of iterations using different seeds. We consider that this is a direct way of estimating the number of evaluations required to get independent posterior samples. Nonetheless, there are some alternative ways of estimating this in the literature.

3.2.7 Effective sample size

The effective sample size (ESS) per posterior evaluation is defined as

$$\text{ESS}(\{\delta_i\}_1^N) = \frac{N}{1 + 2 \sum_{n=1}^{N-1} (1 - \frac{n}{N}) C_n}. \quad (57)$$

Even if the correlation length C_n in theory should asymptotically drop to 0 after a finite number of iterations N , in practice noise will dominate the ESS estimator for large sums. This is why many different solutions have been suggested in the literature (see different definitions in Carpenter et al. 2017; Kaplan 2014; Foreman-Mackey et al. 2013). We decide to follow Carpenter et al. (2017) and Song, Zhao & Ermon (2017) to overcome these problems by truncating the sum over the correlation length when it goes below 0.1. The results of this computation are shown in Figs 10–12. This demonstrates an efficiency of the fourth-order sample over the second-order one of a factor of about 3.0, which is superior to our direct estimation of 2.3 with the CL^{eff} . While our direct CL^{eff} computation assumes a constant CL after convergence (see Section 3.2.2), the ESS estimator does not. The advantage of the CL^{eff} over the ESS is only that it is

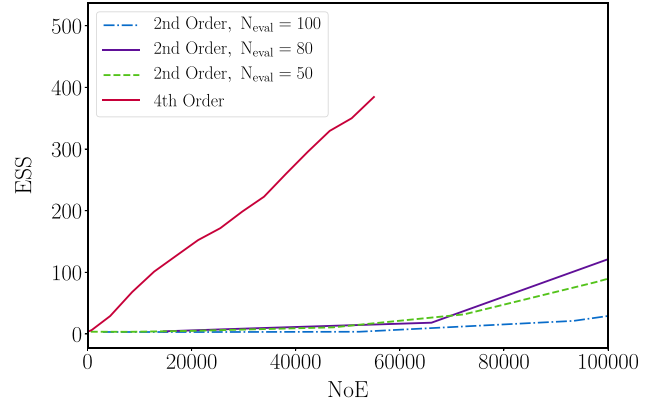


Figure 11. ESS estimator as a function of the number of evaluations of the Hamilton's equation of motion (NoE) for the fourth and the second (with $N_{\text{eval}} = 50, 80, 100$) order schemes.

very fast to compute, and qualitatively yields the same results, as can be seen in Fig. 12. But, in terms of statistical robustness we rather rely on the ESS estimator.

We can conclude from this study that the most efficient way to perform the second-order leap-frog algorithm is to start with a low number of evaluations ($N_{\text{eval}} = 5$) until it reaches convergence, and then change to a high number ($N_{\text{eval}} = 80$) to reduce the correlation length. Our calculations also demonstrate that the fourth-order case delivers the most efficient posterior sampling calculations for high statistical dimensions cases. We have not fully studied the potential of the fourth-order scheme. More efficient fourth-order schemes might be obtained by considering random i within the range 2, ..., 4, or other combinations. Such methods could be improved by also considering different probabilities for each i value. This study is out of the scope of this work.

3.3 Dimensionality and efficiency

To confirm that higher order schemes become more important with increasing number of dimensions in parameter space, we make an additional resolution analysis. This has the additional property of decreasing the uncertainty per dimension (i.e. per cell), since the number density of galaxies increases, and consequently the uncertainty in the matter field (the relative Poisson error) decreases.

The results of this study are shown in Fig. 12. We find that the second-order scheme becomes superior to the fourth-order scheme, going down in resolution to a mesh of 64^3 cells. The fourth-order scheme starts to become moderately more efficient considering meshes of 128^3 cells, and clearly superior for the 256^3 case. This is shown in the left-hand panel in the burn-in phase. The middle and right-hand panels show the consistency between the effective correlation function CL^{eff} and the averaged effective sample size normalized by the respective number of evaluations (ESS/NoE), respectively (after convergence).

Note, that 64^3 corresponds for a volume of $1250 h^{-1} \text{ Mpc}$ to a cell resolution of about $20 h^{-1} \text{ Mpc}$. This is considered a too low resolution for most practical cosmological reconstruction cases (see, e.g. Vargas-Magaña et al. 2017). Current galaxy surveys trace larger cosmic volumes than the one considered in this study making a mesh of 64^3 far from being useful.

By considering larger uncertainties per dimension (i.e. per cell), we can find higher order schemes becoming more efficient than the second-order one. Such a situation corresponds to lowering the

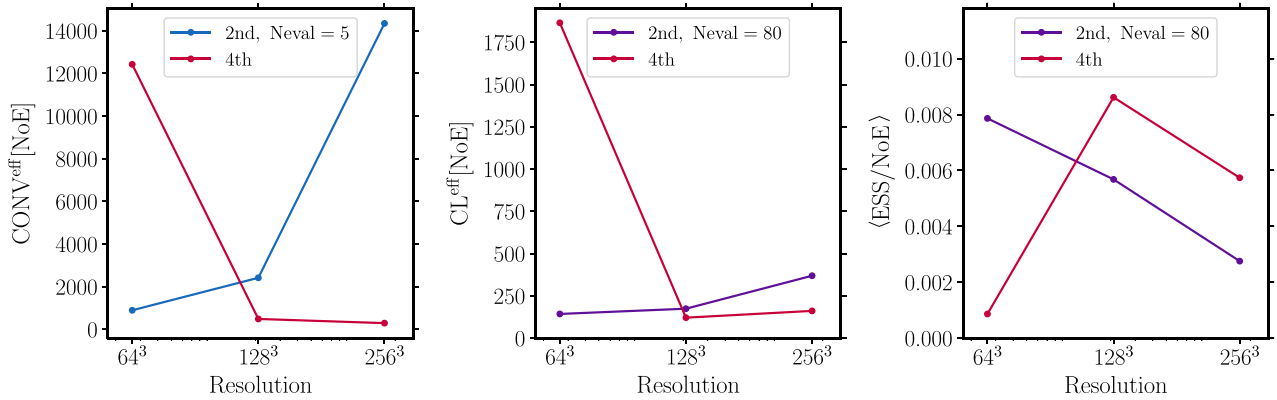


Figure 12. Left-hand panel: average number of evaluations of Hamilton's equations of motion to achieve convergence (CONV^{eff}[NoE]) as a function of resolution (64³, 128³, and 256³ cells), with the red and the blue lines showing the fourth and the second ($N_{\text{eval}} = 5$) order schemes, respectively. Middle panel: correlation length in terms of number of evaluations of Hamilton's equation of motion, also as a function of resolutions. The red and purple lines represent the fourth and second ($N_{\text{eval}} = 80$) order scheme, respectively. Right-hand panel: averaged (from iterations 1000 to 4000) ESS estimator normalized by the number of evaluations of Hamilton's equations of motion (NoE) as a function of resolution. The red and purple lines represent the fourth- and second-order ($N_{\text{eval}} = 80$) schemes, respectively. For each discretization case, the most efficient set-up has been selected. The optimal basic ϵ step sizes were found for the 128³ and 256³ cases. We did not do such a study for the second-order leap-frog 64³ case, since it became already clear that it is superior.

number density (by say an order of magnitude) in the halo catalogue, which implies getting a poorer representation of the underlying dark matter field. However, this becomes in general unrealistic, as we have considered the most massive haloes hosting luminous red galaxies. When considering higher number densities we might find fourth-order schemes to perform worse than the second-order one on meshes of 256³ cells. However, we will be in general interested in covering very large volumes considerably increasing the mesh size. Also, our full reconstruction approach considers tracers in Lagrangian space, where they are more sparse (Kitaura et al. 2020).

In conclusion, it will depend on the particular problem which scheme performs better, but with increasing cosmic volumes second-order schemes will, in general, become a worse choice.

4 CONCLUSIONS

This work presents an efficient Hybrid Markov Chain Hamiltonian Monte Carlo Sampling method for cosmological large-scale structure analysis. In particular, it relies on a fourth-order symplectic integration of Hamilton's equations of motion. This is achieved through an operator formalism in which the original leap-frog algorithm is recursively applied in a combination of two forward time integration steps with an intermediate backward step and appropriate step sizes. One of the key ingredients is to realize that the higher integration accuracy of the fourth-order scheme permits one to fix the number of evaluations of the equations of motions to a few (~ 7), being able to perform larger effective time steps in each evaluation of Hamilton's equations of motion, and obtaining high acceptance rates. At low resolutions (i.e. low dimensional spaces) we find, however, that the second-order scheme is superior, confirming the theoretical expectations (Bou-Rabee & Sanz-Serna 2018). We have restricted this study to the lognormal-Poisson model, applied to a full volume halo catalogue in real space on a cubical mesh of 1250 h^{-1} Mpc, with 128³ and 256³ cells. However, we have shown that selection effects, redshift space distortions, and displacements can be accounted for within a Gibbs-sampling scheme, as implemented in the COSMIC BIRTH algorithm. In this way, the scheme presented here permits one to efficiently sample the primordial density fluctuations of the Universe from galaxy surveys within a posterior Bayesian inference

framework (Kitaura et al. 2020). This scheme can help to improve the efficiency of other Bayesian inference methods (e.g. the publicly available BARCODE Bos et al. 2019).

We have demonstrated performing an extensive parameter study, that going from the usual second to fourth-order in the discretization of Hamilton's equations of motion improves the convergence by a factor of ~ 30 in number of evaluations for the best second-order case. This implies, that 75-90 independent samples are obtained, while the fastest second-order method converges. Moreover, we obtain independent samples about ~ 3.0 times faster than the best second-order scheme, which has a different setting than for the burn-in phase. It is interesting to note, that the most efficient fourth-order case in the burn-in phase is the same, as the one after convergence. This is very convenient, when the ideal set-up of a Hamiltonian sampler for a particular case needs to be investigated. We leave further investigations of more sophisticated higher order schemes for future work (see McLachlan & Quispel (2002) and Blanes et al. (2014), which converges to a 4th order scheme in the limit of vanishing step sizes).

In summary, the investigation of this work shows that improved symplectic integrators can play a major role in gaining computational efficiency for Hamiltonian Monte Carlo sampling methods in high dimensional problems to go towards a full Bayesian analysis of the cosmological large-scale structure for upcoming galaxy surveys.

ACKNOWLEDGEMENTS

The authors thank Jesús Sanz-Serna for explaining them the mathematical reason for the higher efficiency of higher order schemes over the traditional second-order leap-frog scheme towards high dimensions. We also want to thank the STAN team for useful discussions and their interest in higher order schemes although, as it has become clear, the majority of the applications does not require the high dimensionality of the problem considered in this work. The authors thank the anonymous referee, Jorge Martín Camalich, Mattia Dallabrida and Andrés Balaguera-Antolínez for useful comments. MHS thanks the *Astrofísicos Residentes* grant at the IAC for permitting her to work on this study as part of her master thesis presented in July 2018. FSK acknowledges financial support from the Spanish Ministry of Economy and Competitiveness (MINECO) under the Severo Ochoa

program SEV-2015-0548, and for the grants RYC2015-18693 and AAYA2017-89891-P. MA thanks for the hospitality at the IAC and the support from the Kavli IPMU fellowship that permitted him to develop the analysis codes for the ESS estimator, CL, and RG test, used in this work. CDV acknowledges the support of the Ministry of Science, Innovation and Universities (MCIU) through grants RYC-2015-18078 and PGC2018-094975-B-C22.

DATA AVAILABILITY

The data and c++ codes used in this paper will be shared on reasonable request to the corresponding author. The halo catalogue from the BigMD simulation can be obtained at this site: <https://www.cosmosim.org/cms/simulations/bigmdpl/>. The ESS estimator used in this study is publicly available at: https://github.com/gmetin/MCMC/blob/master/corr_length_birth.ipynb. A PYTHON version of the c++ code used in this study is publicly available at: <https://github.com/pacoshu/HMC>, making a comparison of the efficiency between the second- and fourth-order schemes straightforward.

REFERENCES

- Ahn K., Iliev I. T., Shapiro P. R., Srisawat C., 2015, *MNRAS*, 450, 1486
- Anderson L. et al., 2014, *MNRAS*, 441, 24
- Angulo R. E., Baugh C. M., Frenk C. S., Lacey C. G., 2008, *MNRAS*, 383, 755
- Ata M., Kitaura F.-S., Müller V., 2015, *MNRAS*, 446, 4250
- Barp A., Briol F.-X., Kennedy A. D., Girolami M., 2018, *Annu. Rev. Stat. Appl.*, 5, 451
- Behroozi P. S., Wechsler R. H., Wu H.-Y., 2013, *ApJ*, 762, 109
- Beskos A., Pillai N. S., Roberts G. O., Sanz-Serna J. M., Stuart A. M., 2013, *Bernoulli* J., 19, 1501
- Beutler F. et al., 2017, *MNRAS*, 464, 3409
- Blanes S., Casas F., Sanz-Serna J. M., 2014, *SIAM J. Sci. Comput.*, 36, A1556
- Bos E. G. P., Kitaura F.-S., van de Weygaert R., 2019, *MNRAS*, 488, 2573
- Bose B., Koyama K., 2017, *J. Cosmol. Astropart. Phys.*, 2017, 029
- Bou-Rabee N., Sanz-Serna J. M., 2018, *Acta Numer.*, 27, 113
- Campostrini M., Rossi P., 1990, *Nucl. Phys. B*, 329, 753
- Carpenter B. et al., 2017, *J. Stat. Softw.*, 76, 1
- Carron J., Szapudi I., 2014, *MNRAS*, 439, L11
- Chao W.-L., Solomon J., Michels D., Sha F., 2015, in Bach F., Blei D., eds, *Proc. 32nd Int. Conf. Machine Learning Research Vol. 37*, PMLR, Lille, France, p. 1142
- Chuang C.-H. et al., 2015, *MNRAS*, 452, 686
- Chuang C.-H. et al., 2017, *MNRAS*, 471, 2370
- Coles P., Jones B., 1991, *MNRAS*, 248, 1
- Creutz M., 1988, *Phys. Rev. D*, 38, 1228
- Creutz M., Gocksch A., 1989, *Phys. Rev. Lett.*, 63, 9
- Duane S., Kennedy A., Pendleton B. J., Roweth D., 1987, *Phys. Lett. B*, 195, 216
- Eisenstein D. J., Seo H.-J., Sirko E., Spergel D. N., 2007, *ApJ*, 664, 675
- Foreman-Mackey D., Hogg D. W., Lang D., Goodman J., 2013, *PASP*, 125, 306
- Gelman A., Rubin D. B., 1992, *Stat. Sci.*, 7, 457
- Gil-Marín H., Percival W. J., Verde L., Brownstein J. R., Chuang C.-H., Kitaura F.-S., Rodríguez-Torres S. A., Olmstead M. D., 2017, *MNRAS*, 465, 1757
- Hairer E., Lubich C., Wanner G., 2010, *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer, Berlin
- Hashimoto I., Rasera Y., Taruya A., 2017, *Phys. Rev. D*, 96, 043526
- Hoffman M. D., Gelman A., 2014, *J. Mach. Learn. Res.*, 15, 1593
- Islas A., Schober C., 2004, *J. Comput. Phys.*, 197, 585
- Jasche J., Kitaura F. S., 2010, *MNRAS*, 407, 29
- Jasche J., Lavaux G., 2019, *A&A*, 625, A64
- Jasche J., Wandelt B. D., 2013, *MNRAS*, 432, 894
- Jasche J., Kitaura F. S., Li C., Enßlin T. A., 2010, *MNRAS*, 409, 355
- Kaplan D., 2014, *Bayesian Statistics for the Social Sciences*, 1st edn. Guilford Press, New York
- Kennedy A. D., 2006, Kuramashi Y., *Perspectives in Lattice QCD*, University of Tsukuba, Japan, preprint ([arXiv:hep-lat/0607038](https://arxiv.org/abs/hep-lat/0607038))
- Kitaura F. S., Enßlin T. A., 2008, *MNRAS*, 389, 497
- Kitaura F.-S., 2013, *MNRAS*, 429, L84
- Kitaura F.-S., Angulo R. E., 2012, *MNRAS*, 425, 2443
- Kitaura F.-S., Jasche J., Metcalf R. B., 2010, *MNRAS*, 403, 589
- Kitaura F.-S., Gallerani S., Ferrara A., 2012a, *MNRAS*, 420, 61
- Kitaura F.-S., Erdoğan P., Nuza S. E., Khalatyan A., Angulo R. E., Hoffman Y., Gottlöber S., 2012b, *MNRAS*, 427, L35
- Kitaura F.-S., Yepes G., Prada F., 2014, *MNRAS*, 439, L21
- Kitaura F.-S. et al., 2016, *Phys. Rev. Lett.*, 116, 171301
- Kitaura F.-S., Ata M., Rodríguez-Torres S. A., Hernández-Sánchez M., Balaguera-Antolínez A., Yepes G., 2020, *MNRAS*, preprint ([arXiv:1911.00284](https://arxiv.org/abs/1911.00284))
- Libeskind N. I. et al., 2018, *MNRAS*, 473, 1195
- Luscher M., 2010, *Modern Perspectives in Lattice QCD: Quantum Field Theory and High Performance Computing*. Les Houches, France, 331, p. preprint ([arXiv:1002.4232](https://arxiv.org/abs/1002.4232))
- McLachlan R. I., 2002, *Numer. Algorithms*, 31, 233
- McLachlan R. I., Quispel G., 2002, *Acta Numer.*, 11, 341
- Mannseth J., Kleppe T. S., Skaug H. J., 2016, *On the application of higher order symplectic integrators in Hamiltonian Monte Carlo*. preprint ([arXiv:1608.07048](https://arxiv.org/abs/1608.07048))
- Mo H., van den Bosch F. C., White S., 2010, *Galaxy Formation and Evolution*, Cambridge Univ. Press, New York
- Neal R. M., 1993, *Tech. Report CRG-TR-93-1*, Probabilistic Inference Using Markov Chain Monte Carlo Methods, Univ. Toronto, 144pp.
- Neal R. M., 2012, preprint ([arXiv:1206.1901](https://arxiv.org/abs/1206.1901))
- Neyrinck M. C., Szapudi I., Szalay A. S., 2009, *ApJ*, 698, L90
- Neyrinck M. C., Aragón-Calvo M. A., Jeong D., Wang X., 2014, *MNRAS*, 441, 646
- Nishimichi T. et al., 2009, *PASJ*, 61, 321
- Okumura T., Hand N., Seljak U., Vlah Z., Desjacques V., 2015, *Phys. Rev. D*, 92, 103516
- Omelyan I. P., Mryglod I. M., Folk R., 2002, *Phys. Rev. E*, 66, 026701
- Padmanabhan N., Xu X., Eisenstein D. J., Scalzo R., Cuesta A. J., Mehta K. T., Kazin E., 2012, *MNRAS*, 427, 2132
- Peebles P. J. E., 1980, *The Large-Scale Structure of the Universe*, Princeton Univ. Press, Princeton, NJ.
- Reid B. A., White M., 2011, *MNRAS*, 417, 1913
- Rein H., Tamayo D., 2018, *MNRAS*, 473, 3351
- Rodríguez-Torres S. A. et al., 2016, *MNRAS*, 460, 1173
- Ross A. J. et al., 2017, *MNRAS*, 464, 1168
- Saito S., Baldauf T., Vlah Z., Seljak U., Okumura T., McDonald P., 2014, *Phys. Rev. D*, 90, 123522
- Saslaw W. C., 1989, *ApJ*, 341, 588
- Schmittfull M., Feng Y., Beutler F., Sherwin B., Chu M. Y., 2015, *Phys. Rev. D*, 92, 123522
- Schuhmann R. L., Joachimi B., Peiris H. V., 2016, *MNRAS*, 459, 1916
- Sheth R. K., 1998, *MNRAS*, 299, 207
- Song J., Zhao S., Ermon S., 2017, preprint ([arXiv:1706.07561](https://arxiv.org/abs/1706.07561))
- Souradeep T., Das S., Wandelt B., 2016, *J. Phys.: Conf. Ser.*, 759, 012062
- Springel V., 2005, *MNRAS*, 364, 1105
- Taylor J. F., Ashdown M. A. J., Hobson M. P., 2008, *MNRAS*, 389, 1284
- Uhlemann C., Kopp M., 2015, *Phys. Rev. D*, 91, 084010
- Vargas-Magaña M., Ho S., Fromenteau S., Cuesta A. J., 2017, *MNRAS*, 467, 2331
- Wang H., Mo H. J., Yang X., van den Bosch F. C., 2013, *ApJ*, 772, 63
- Wang H., Mo H. J., Yang X., Jing Y. P., Lin W. P., 2014, *ApJ*, 794, 94
- White M., 2015, *MNRAS*, 450, 3822
- White M., Tinker J. L., McBride C. K., 2014, *MNRAS*, 437, 2594
- Yoshida H., 1990, *Phys. Lett. A*, 150, 262
- Zaroubi S., Hoffman Y., Fisher K. B., Lahav O., 1995, *ApJ*, 449, 446
- Zhao C. et al., 2020, *MNRAS*, 491, 4554

This paper has been typeset from a \LaTeX file prepared by the author.